

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
14 February 2002 (14.02.2002)

PCT

(10) International Publication Number
WO 02/13470 A2

(51) International Patent Classification⁷: **H04L 12/58**

(21) International Application Number: PCT/US01/25034

(22) International Filing Date: 8 August 2001 (08.08.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/634,343 8 August 2000 (08.08.2000) US

(71) Applicant: **TUMBLEWEED COMMUNICATIONS CORP.** [US/US]; 700 Saginaw Drive, Redwood City, CA 94063 (US).

(72) Inventors: **BANDINI, Jean-Christophe**; 3375 Brittan Avenue, #11, San Carlos, CA 94070 (US). **BEER, John, Kenneth**; 20 Elm Street, San Carlos, CA 94070 (US). **NEVEU, Jean-Sebastian, T.**; 3131 Homestead Road, Apt. 19-H, Santa Clara, CA 95051 (US). **SMITH, Jeffrey, C.**; 158 Almendral Avenue, Atherton, CA 94027 (US).

(74) Agent: **IVEY, James, D.**; Law Offices of James D. Ivey, 3025 Totterdell Street, Oakland, CA 94611-1742 (US).

(81) Designated States (*national*): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.

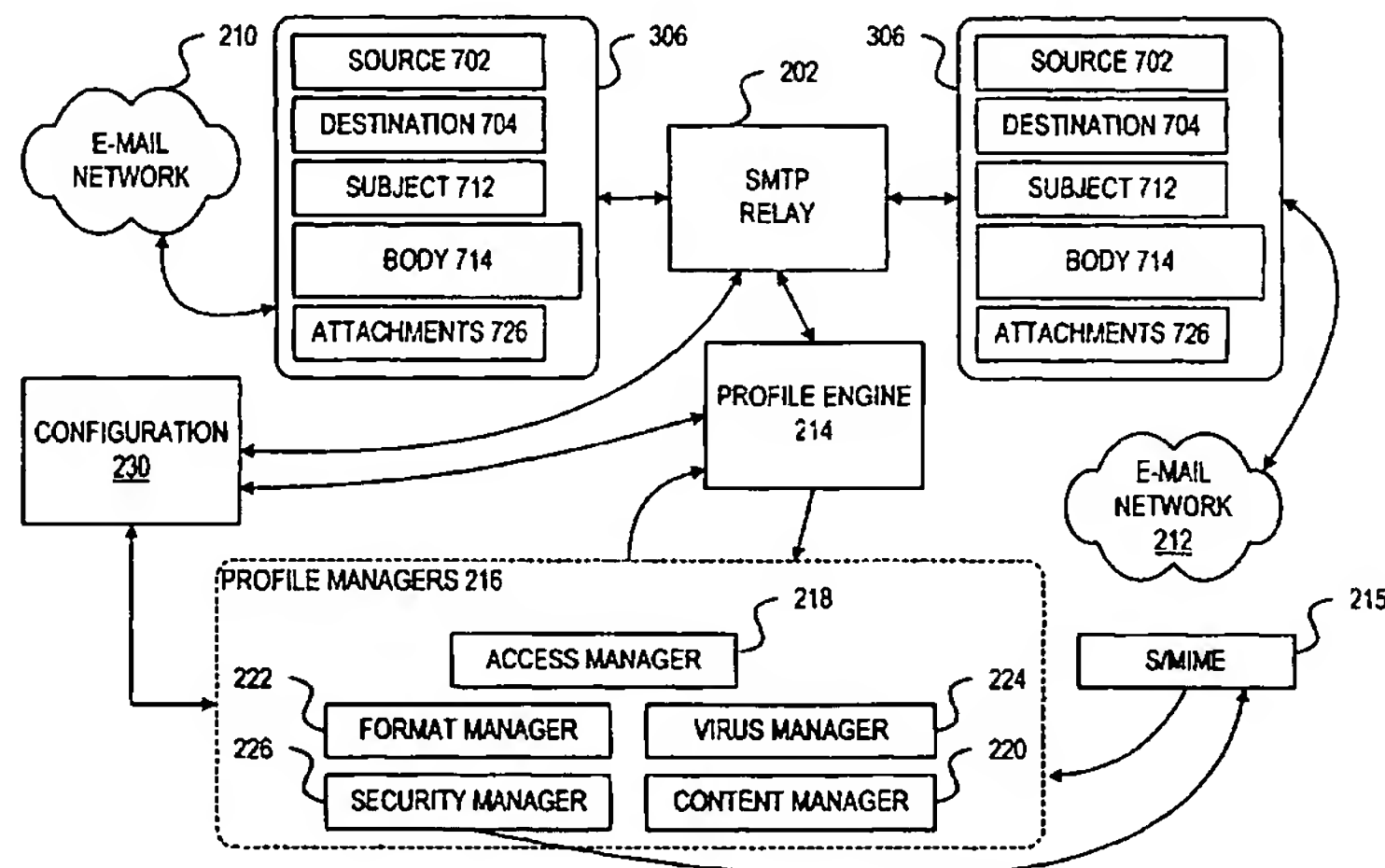
(84) Designated States (*regional*): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: RECIPIENT-SPECIFIED AUTOMATED PROCESSING OF ELECTRONIC MESSAGES



(57) Abstract: An e-mail relay receives a message, processes the message in a manner specified by a recipient of the message, and forwards the message to the recipient at a remotely located computer if such processing indicates that delivery of the message is to proceed. The message can include a message body and can also include one or more attached data files. The e-mail relay is intended to be always available and operational such that recipient-specified processing of incoming messages is immediate notwithstanding unavailability of the computer used by the recipient to receive and view incoming messages. Processing on behalf of the recipient by the e-mail firewall is specified by a profile associated with the recipient. The profile associates one or more conditions with one or more actions to be performed on behalf of the recipient if the associated conditions are met.



WO 02/13470 A2

RECIPIENT-SPECIFIED AUTOMATED PROCESSING OF ELECTRONIC MESSAGES

SPECIFICATION

This is a continuation-in-part of International Patent Application PCT/US98/15552 for which the United States is a designated state and which claims priority of U.S. Patent Application S/N 60/053,668.

FIELD OF THE INVENTION

The invention relates to data transfer through computer networks and, in particular, to a mechanism by which packages to be delivered to a recipient are automatically processed in a manner specified by the recipient.

BACKGROUND OF THE INVENTION

The Internet has grown tremendously in recent years, both in terms of number of users and the amount of data transferred through the Internet. Originally, the Internet was a data transfer medium for academia. Eventually, engineers and private users increasingly used and became more familiar with the Internet. More and more, the Internet is becoming an acceptable communication medium for business. However, business users demand more confidentiality and traceability of communication than do private users engaging in personal correspondence.

Business users often communicate sensitive, confidential, and proprietary information and, accordingly, expect such communication to be secure from unauthorized eavesdropping. In addition, business users expect to be able to store records tracing correspondence. Accordingly, to provide a medium for business communication, Internet-based communication must be made secure and traceable.

While security and traceability of Internet-based communication is improving, many of the conveniences of conventional paper mail is still lacking in current Internet

messaging systems. For example, a business person can instruct support staff to examine contents of the business person's in-box and forward any items from a particular party to another person for special attention. Current e-mail clients, whether web-based thin clients or thick clients, are generally limited to conditional processing based on textual analysis of the contents of the message and result in sorting the message into one of a number of folders. Such is generally inadequate to satisfy the ever increasing demands for information processing imposed upon today's businesses. In addition, such conventional e-mail clients typically require that the recipient's computer is currently running and has retrieved incoming messages to filter such messages.

What is needed is a mechanism for improved recipient-specified automated processing of electronic messages and attached data files.

SUMMARY OF THE INVENTION

In accordance with the present invention, an e-mail relay receives a package through a first computer network, processes the package in a manner specified by a recipient of the package, and forward the package to the recipient through a second computer network if such processing indicates that delivery of the package is to proceed. The package can include a message and can also include one or more attached data files.

Recipient-specified processing in accordance with the present invention provides a significant advantage over conventional filtering within an e-mail reader. Typically, a conventional e-mail reader can only filter incoming e-mail messages when (i) the recipient's computer is running and is connected to the network through which e-mail is received, (ii) the e-mail reader is executing within the recipient's computer, and (iii) the e-mail reader has already retrieved the recipient's e-mail. In contrast, an e-mail relay in accordance with the present invention is intended to be always available and operational such that recipient-specified processing of incoming packages is immediate notwithstanding unavailability of the computer used by the recipient to receive and view incoming packages. For example, if a package is received during off hours (e.g., very early in the morning), the package can be processed in a manner specified by the recipient even if the recipient's computer is unavailable — for example, by sending an automated

reply to the package and/or sending a copy of the package to another party. The sender therefore doesn't have to wait for several hours for handling of the package to begin.

The processing of incoming package can be specified by the recipient as a list of associations between one or more conditions and one or more actions to be carried out upon satisfaction of the associated conditions. Each condition includes a boolean expression involving one or more sender attributes, recipient attributes, package attributes, and/or environmental attributes. Sender and recipient attributes can include regular expressions involving e-mail addresses by which each is specified or can include attributes of user records specifying each. User record attributes can be particularly useful in categorizing the sender and the recipients as belonging to particular divisions within an organization, although it is appreciated that e-mail addresses can sometimes provide similar information.

Package attributes include message attributes, delivery attributes, and attached data files. Message attributes include a subject and a message body. Conditions involving message attributes can be used to detect urgent or particularly important information in a package and/or inappropriate or offensive content such as sexually explicit language and/or unwanted solicitations and advertisements. Delivery attributes include such things as package delivery priority, security options, and delivery timing. Conditions involving delivery attributes can detect packages which have a particularly high priority and therefore warrant special handling, for example. Attached data files can include confidential information, can include inappropriate material, can be excessive in size, and can include malicious computer instructions in the form of viruses or Trojan horses for example. Conditions can detect specific conditions of data files attached to the package.

Actions can interrupt delivery of the package, log handling of the package, or modify the package. Examples of interrupting actions include discarding the package or redirecting the package to a different recipient. Logging actions can include, for example, saving a copy of the package, sending a copy of the package to a predetermined recipient, and notifying a predetermined entity (perhaps the sender) of another action taken with respect to the package. Modification actions can modify the package by changing the subject, the message, delivery attributes, and the attached data files. For example, all attached data files can be removed or only those attached data files which satisfy a

particular set of conditions can be removed.

BRIEF DESCRIPTION OF THE DRAWINGS

Figure 1 is a block diagram of a delivery system in accordance with the present invention.

Figure 2 is a block diagram of the e-mail firewall of Figure 1 in accordance with the present invention.

Figure 3 is a block diagram of the profile and message datastore of Figure 1 in greater detail.

Figure 4 is a logic flow diagram of the enforcement of profiles by the e-mail firewall of Figure 2 in accordance with the present invention.

Figure 5 is a block diagram of a profile in accordance with the present invention.

Figure 6 is a block diagram of a rule of the profile of Figure 5 in greater detail.

Figure 7 is a block diagram of a package of Figure 2 in greater detail.

Figure 8 is a block diagram illustrating recipient-specific processing.

Figure 9 is a block diagram illustrating interrelationships of conditions through boolean operators.

Figure 10 is a block diagram showing an action of the rule of Figure 6 in greater detail.

Figure 11 is a block diagram showing an alternative embodiment of the action of the rule of Figure 6.

Figure 12 is a block diagram of a user record.

DETAILED DESCRIPTION

In accordance with the present invention, packages sent to a recipient through computer network 104 are automatically processed in a manner specified by the recipient. The package is transferred by a sender, using computer system 112 for example, through computer network 104 to an e-mail firewall 105.2 for delivery through e-mail network 103 to the recipient, using computer system 114 for example. It should be noted that, while e-

mail firewall 105.2 is shown as positioned between computer network 104 and e-mail network 103, e-mail firewall 105.2 can be positioned in any manner that permits e-mail firewall 105.2 to intercept packages intended for the recipient for processing in the manner described herein. E-mail firewall 105.2 temporarily stores the package in a datastore 120 and processes the package in a manner described more completely below in accordance with policies specified by the recipient. Delivery of packages through the system of Figure 1 is described more completely in International Patent Application PCT/US98/15552 for which the United States is a designated state, and that description is incorporated herein by reference. As described more completely below, a package is a collection of computer-readable data which can include a message body and can also include zero or more attached data files. In one illustrative embodiment, a package is a standard e-mail message which can be transported through the Internet according to the Simple Mail Transport Protocol (SMTP).

In this illustrative embodiment, automated processing on behalf of the recipient is performed by e-mail firewall 105.2. A sender sends a package containing a message and zero or more data files from a computer 102, through computer network 104 to a computer 114 through which the recipient ultimately receives the package. The package is delivered to computer 114 through e-mail network 103 by e-mail firewall 105.2. In this illustrative embodiment, computer network 104 is a publicly-accessible wide area network such as the Internet and e-mail network 103 is a private, secure local area network. However, it is appreciated that computer networks 103 and 104 can be any computer network and, in fact, can be the same network. What is significant is that packages directed to the recipient are intercepted by e-mail firewall 105.2 such that processing on behalf of the recipient can be performed by e-mail firewall 105.2.

A number of advantages are provided by the automated package processing mechanism described herein. For example, automated processing is available notwithstanding unavailability of computer 114 of the recipient. In many e-mail readers which perform filtering, such filtering can only be performed when the receiving computer is running and e-mail is retrieved. Any messages sent during the night or a weekend or while the recipient is vacationing, for example, cannot be filtered until the recipient boots the receiving computer and connects to the network for access to e-mail messages. In

contrast, e-mail firewall 105.2 is intended to execute continuously, regardless of whether computer 114 is currently running and accessible. Accordingly, automated processing on behalf of the recipient can proceed notwithstanding inaccessibility of computer 114. Such is particularly useful in time-based conditional processing described more completely below.

As mentioned herein, the package addressed to the recipient can include zero or more attached data files. A data file can contain any type of computer-readable data such as text, graphical images, motion video, audio signals, database records, etc. Data files can be stored in any of a number of computer-readable storage media and can be transferred through a computer network such as computer network 104.

As described briefly above, datastore 120 stores packages to be delivered according to package delivery system 100. Datastore 120 is shown in greater detail in Figure 3. Datastore 120 includes a number of user records, each representing an entity capable of receiving packages through e-mail firewall 105.2 (Figure 1). For example, user record 304 (Figure 3) can represent the intended recipient of a package such as package 306. As used herein, a user of e-mail firewall 105.2 can be either a human user or a computerized user. A computerized user is all or part of one or more computer processes and can send and/or receive packages through e-mail firewall 105.2. It should be understood that, like the recipient, the sender and any other recipients described in this illustrative embodiment can be either human users or computerized users.

Each user record, e.g., user record 304, is associated with one or more profile records 308. Each profile record 308 represents a number of rules to be applied to packages addressed to the user represented by the associated user record 304. The manner in which rules are applied is described more completely below. In addition, each profile record, e.g., profile record 208, can be associated with more than one user. For example, a single user can have multiple e-mail addresses and can specify that all profiles are to be applied to all e-mail addresses of the user. In addition, profile records can be associated with groups of users, and the groups of users can be organized hierarchically such that profiles are inherited but can be superseded by profiles associated with lower levels of the user group hierarchy.

A user, referred to as the sender, can create one or more packages for delivery to

another user, e.g., the recipient. In this illustrative embodiment, each such package is a standard e-mail message and can include one or more attached data files. Each such package is represented by a package record such as package 306. The profiles represented by profile records 308 are applied to all packages addressed to the user of user record 304.

E-mail firewall 105.2 includes an SMTP relay 202 (Figure 2), a profile engine 214, profile managers 216, and a configuration 230 data structure which includes user record 304 (Figure 3) and profile record 308. It should be noted that, while e-mail firewall 105.2 is shown as positioned between computer network 110 and e-mail network 103, e-mail firewall 105.2 can be positioned in any manner that permits e-mail firewall 105.2 to intercept packages intended for the recipient for processing in the manner described herein. It should also be appreciated that e-mail firewall 105.2 can be implemented using several computer systems which cooperate with one another, perhaps through a computer network, to provide the services described herein. Connectivity of such distributed processes is conventional and known and can be implemented using such standard and conventional techniques as sockets, pipes, remote procedure call (RPC), Common Object Request Broker Architecture (CORBA), Distributed Object Component Model (DCOM), and Java Remote Method Invocation (Java RMI). To provide the services described herein in an efficient manner, it is preferred that the various distributed components of e-mail firewall 105.2 interact through relatively quick, efficient channels, e.g., with low latency and relatively high bandwidth.

The sender creates a package using a conventional e-mail composer and sends the package through computer network 104 according to SMTP (Simple Mail Transfer Protocol). A package created by the sender as represented by package 306 (Figure 3) is described in greater detail below in conjunction with Figure 7. Briefly, the package includes address data specifying one or more recipients, subject and message data, delivery and post handling specification data, and can include one or more attached data files.

SMTP relay 202 of e-mail firewall 105.2 receives package 306 which is addressed to the subject recipient as described briefly above. Before forwarding package 306 to the recipient, SMTP relay 202 submits package 306 to a profile engine 214 for processing in a manner specified by the recipient. Package 306 can be addressed to more than one recipient as illustrated in Figure 8. Accordingly, profile engine 214 processes package 306

according to the respective profiles for each recipient. Figure 8 shows prioritized lists 804, 806, and 808 of profile rules for first, second, and third recipients respectively. Profile engine 214 processes package 306 according to each list of profile rules independently such that the profile for a particular recipient applies only to that recipient and to no other recipients of package 306. As shown in Figure 8, profile engine 214 can also process profile rules established by the sender.

Profile engine 214 performs recipient-specified processing for a particular recipient as illustrated by logic flow diagram 400 (Figure 4). In test step 402, profile engine 214 (Figure 3) determines whether any profiles exist for the recipient. In particular, profile engine 214 determines the recipient either from the package itself, e.g., package 306 (Figure 4). In addition, profile engine 214 (Figure 3) retrieves the recipient's user record 304 (Figure 3). User record 304 can include pointers to one or more profile records such as profile record 308 or, alternatively, nil to indicate that no profiles are established for the recipient.

While a simple, direct association between the recipient and profiles applicable to the recipient is shown, it is appreciated that more complex associations can be implemented. One example is a hierarchical grouping of users such that profiles associated with groups are inherited by members of the group or subgroups thereof. In such an embodiment, profiles can be specified for individual members or for subgroups such that inherited profiles are superseded. Many other mechanisms can be used to associate profiles directly or indirectly with individual recipients.

If no profiles are established for the recipient, e.g., if user record 304 includes nil data to indicate no such profiles are established, processing by profile engine 214 (Figure 2) transfers to terminal step 404 (Figure 4) in which the sending of the subject package is resumed and processing according to logic flow diagram 400 terminates.

If one or more profiles are established for the recipient, processing transfers to loop step 406. Loop step 406 and next step 412 define a loop in which each rule of each profile is processed according to steps 408-410. Each profile includes one or more rules. For example, profile record 308 (Figure 3 and shown in greater detail in Figure 5) includes one or more rule records 502. Each rule record, e.g., rule record 502, includes one or more condition records 602 (Figure 6) and one or more action records 604. During a particular

iteration of the loop of steps 406-412 (Figure 4), the rule processed according steps 408-410 is referred to herein as the subject rule.

In test step 408, profile engine 214 (Figure 2) determines whether the conditions represented by the condition records of the subject rule are collectively met by the subject package. As described in greater detail below, each condition, e.g., condition record 602 (Figure 6), specifies a boolean expression involving an attribute of a package or a package-independent attribute. Various attributes of a package are described in greater detail below. Briefly, such attributes can include the sender, one or more recipients, the subject, the message body, delivery attributes, post handling procedures, and one or more attached files. Package-independent attributes include, for example, date/time, time since or before an event, and recipient-specified state variables such as 'on vacation' and 'traveling on business.' Boolean expressions involving the sender and/or recipients of a package can specify all or part of e-mail addresses, for example using a regular expression. One possible use of e-mail addresses in a condition would be to distinguish recipients within an organization of which the sender is a member from recipients outside such an organization. Boolean expressions involving textual attributes such as subject and message body and textual content of attached data files can be used to search for inappropriate terms which can be offensive to the recipient or which indicate that the package is an unwanted solicitation or can indicate an urgent matter requiring immediate attention. Boolean expressions involving delivery attributes can be used to direct the package to an appropriate destination device. Boolean expressions involving attached data files can be used to detect the spread of malicious programs and to discriminate between professional and personal correspondence.

The conditions of a rule, e.g., all condition records 602 of rule record 502, are related to one another through boolean operators. Figure 9 shows a tree structure 900 in which a number of conditions, e.g., conditions 602 and 602B-E, are related to one another by boolean operators 902-908. In this illustrative example, (i) boolean operator 902 specifies a logical "OR" relation between conditions 602 and 602B-C, (ii) boolean operator 904 specifies a logical "AND" relation between conditions 602D-E, (iii) boolean operator 906 specifies a logical negation of the intermediate result of boolean operator 904, and (iv) boolean operator 908 specifies a logical "AND" relation between the

intermediate result of boolean operator 902 and the intermediate result of boolean operator 906.

If profile engine 214 (Figure 2) determines that the conditions of the subject rule, collectively in accordance with logical relations to one another, are not satisfied by the subject package, processing transfers from test step 408 (Figure 4) to next step 412, skipping step 410, and the next rule is processed according to the loop of steps 406-412. Conversely, if profile engine 214 (Figure 2) determines that the conditions of the subject rule, collectively in accordance with logical relations to one another, are satisfied by the subject package, processing transfers from test step 408 (Figure 4) to step 410.

In step 410, profile engine 214 (Figure 2) adds all actions for the subject rule to a list of actions for the subject package. This list is initialized to be empty upon initiation of processing according to logic flow diagram 400 (Figure 4) and at least prior to processing according to the loop of steps 406-412. While performance of actions of the list is postponed in this illustrative embodiment until conditions of all rules of a profile have been tested, such actions are performed immediately in an alternative embodiment. After step 410, processing transfers to next step 412 and the next rule is processed according to the loop of steps 406-412. Once all rules of all profiles of the recipient have been processed according to the loop of steps 406-412 (Figure 4), processing transfers to step 414.

In step 414, profile engine 214 (Figure 2) orders the actions of the list of actions according to priority. Some actions work better if performed before other actions. For example, if an action modifies the body of a message of a package and another action forwards a copy of the package to a predetermined recipient, it is preferred that the copy include the modified body. In other words, it is preferred that the modification action precedes the forwarding action.

Figure 10 shows action 604 in greater detail. Action 604 includes an action body 1002 which specifies the specific action to be taken when performing action 604, and a priority 1004. Priority 1004 is established by the recipient and, in step 414 (Figure 4), profile engine 214 (Figure 2) sorts actions of the list such that higher priority actions are performed before actions of lower priority.

Figure 11 shows an action 604B in accordance with an alternative embodiment.

Action 604B includes action data 1102 and a reference 1104 to an action definition 1106. Action data 1102 specifies data relevant to the action represented by action 604B. For example, if action 604B specifies that a copy of the package is to be forwarded, action data 1102 can specify an e-mail address to which the copy is forwarded. Action definition 1106 specifies the details of the action to be taken and includes a priority 1108. Priority 1108 is established by the recipient and, in step 414 (Figure 4), profile engine 214 (Figure 2) sorts actions of the list such that higher priority actions are performed before actions of lower priority.

After step 414 (Figure 4), processing transfers to step 416. The list of actions to be performed with respect to the subject package can contain duplicate, redundant actions. For example, a single package can satisfy more than one set of conditions thereby potentially adding identical actions to the list of actions. Accordingly, profile engine 214 (Figure 2), in step 416 (Figure 4), removes duplicate actions from the list of actions to perform with respect to the subject package. Thus, each action is performed only once for the subject package.

It is appreciated that the relative order of steps 414-416 is not important. For example, step 416 can be performed before step 414. In addition, actions associated with satisfied conditions are performed in step 410 above in an alternative embodiment as described above. In this alternative embodiment, steps 414-422 are obviated.

Loop step 418 and next step 422 define a loop in which profile engine 214 (Figure 2) performs each of the actions of the list of actions in step 420 (Figure 4) for the subject package. Actions performed by profile engine 214 (Figure 2) in step 420 (Figure 4) and specified by action record 604 (Figure 6) generally affect the delivery of the subject package. While there are many types of actions which can affect delivery of the subject package, three (3) major categories are particularly useful in conjunction with the illustrative embodiment described herein. In particular, actions can (i) direct delivery of the package, (ii) distribute information regarding the package, and/or (iii) modify the package.

Actions Specified by the Recipient

Actions in the first category include discarding the package and automatically

replying to the sender of the package. Profile engine 214 (Figure 2) can discard a package by deleting the package record representing the discarded package, e.g., package 306, or alternatively, by removing the recipient for which the package is discarded from a list of recipients of the package. Profile engine 214 automatically replies to the sender by creating a new package with a message body previously specified by the recipient and sending the new package to the sender.

Actions by profile engine 214 which distribute information regarding a package include redirecting the package to a different recipient, sending a copy of the package to a different recipient, and notifying a different recipient of the existence of the package. Profile engine 214 redirects a package by substituting a new recipient for the original recipient. The package is then delivered to the new recipient and the original recipient of the package never receives the package. Such is useful if the original recipient has shed a responsibility assumed by another. Packages pertaining to such a responsibility, e.g., handling vacation requests within a department of a company, can be redirected to the person currently handling that responsibility without cluttering the in-box of the original recipient. Profile engine 214 can send a copy of a package to a predetermined entity, e.g., at a predetermined e-mail address. The e-mail address can correspond to a third party interested in the type of package received or can be an alternate e-mail address for the recipient, e.g., an alphanumeric pager. Notification of a package is similar to sending a copy of the message except that the notification can include less than the entirety of the package, e.g., the subject only or the subject and message with no attached data files, or can contain no text of the package itself. For example, the notification can be a previously specified textual message indicating that a package has been received. The notification message can include some information regarding the received package such as the sender of the package and the priority of the package. As with sending a copy of a package, the notification message can be sent to an e-mail address of an interested third party and/or an alternative e-mail address of the original recipient, e.g., to an alphanumeric pager, wireless telephone with text messaging capability such as Short Message Service (SMS) or Wireless Application Protocol (WAP), printer, fax machine, etc.

Profile engine 214 can perform actions which modify the package. Actions which modify the package modify one or more of the fields of the package. For example, an

action can prepend or append text to the message body of the package, can remove all attached data files or those attached data files which satisfy the conditions of the rule, and can modify parameters of the package such as the package's priority. In addition, actions can modify the package by removing malicious computer instructions from one or more attached data files, can compress or decompress one or more attached data files, and can initiate execution one or more computer processes while supplying one or more attached data files to the one or more computer processes as input data. The latter action allows new actions to be developed subsequently and used to process attached data files without requiring creation of new actions recognized by and applied by profile engine 214 (Figure 2).

After all actions of the list of actions have been performed by profile engine 214 in the loop of steps 418-422 (Figure 4), processing transfers to test step 424. In test step 424, profile engine 214 (Figure 2) determines whether the subject package is ready to be delivered. In this illustrative embodiment, all packages which are not discarded or redirected are ready to be delivered. If the subject package is discarded or redirected, processing transfers to terminal step 428 (Figure 4) in which processing according to logic flow diagram 400 terminates and the subject package is not delivered. Conversely, if the subject package is ready to be delivered, processing transfers to terminal step 426 in which processing according to logic flow diagram 400 terminates and delivery of the subject package in the manner described above continues.

Conditions Based on Package Structure

As described above, packages are addressed from a sender to one or more recipients and can include a message and one or more attached data files. In addition, profile conditions include boolean expressions involving attributes of a package, and application of a rule can modify a package. A package, e.g., package 306 (Figure 3), is shown in greater detail in Figure 7.

Package 306 includes a sender field 702. A condition such as condition 602 (Figure 6) can include boolean expressions involving sender field 702 (Figure 7). For example, condition 602 can include a regular expression which can match one or more e-mail addresses. Regular expressions are well-known and are not described herein.

Regular expressions are considered a type of boolean expression in which a matching condition is equivalent to a “true” boolean value and a non-matching condition is equivalent to a “false” boolean value.

Condition 602 can also include a boolean expression involving an attribute of a user record, e.g., user record 304, corresponding to the sender specified in sender field 702. To detect such a condition, package and account datastore 120 includes data mapping various e-mail addresses to corresponding user records such as user record 304 in one embodiment. In addition, profile engine 214 (Figure 2) can access data representing one or more characteristics of the sender from an external directory 122 (Figure 1) which is accessible through computer network 110. Directory 122 is an external directory accessible according to a database access protocol such as the X.500 Standard or a similar directory service such as Lightweight Directory Access Protocol (LDAP), NetWare Directory Service (NDS), and Active Directory. In addition, information regarding the sender can be obtained from external databases which are not traditionally considered user directories.

Directories are databases which are intended to be read more often than written such that data stored therein is relatively stable and unchanging. Similar data can be stored in a regular database as well. Profile engine 214 (Figure 2) is configured to retrieve data representing one or more characteristics of the sender from a database in which data in sender field 702 (Figure 7) is directly or indirectly associated with data representing the sender. Such databases can include, for example, human resources databases, customer care and support databases, and enterprise resource planning (ERP) databases such as that provided by SAP AG (see, e.g., <http://www.sap.com/> on the World Wide Web). Such databases include information about users and provide some mechanism for accessing such information — typically an Applications Programming Interface (API).

User record 304 or external user databases can include classification data which is useful in determining membership of the sender in any of a number of groups. For example, a condition can include a boolean expression which determines whether the sender is in the legal department or sales department of the same company as the recipient or works in a specific office of the company, e.g., a Japanese branch office. By testing for particular senders and/or classes of senders, the profile engine 214 can apply different

rules to various senders.

User record 304 is shown in greater detail in Figure 12. Each of the fields of user record 304 can be used in a condition such as condition 602 (Figure 6) to identify packages from a particular sender or a particular class of senders. User record 304 (Figure 12) includes a name field 1202, an e-mail address 1204, a group field 1206, a title field 1208, an organization field 1210, an account field 1212, a database identifier 1214, a schedule pointer 1216, user-defined state variables 1218, and public key 1220. Name field 1202 specifies the name of the user represented by user record 304, referred to as the subject user in the context of Figure 12. E-mail address 1204 specifies the e-mail address of the subject user. Group 1206 specifies one or more groups or departments to which the subject user belongs. Title field 1208 specifies the title or position held by the subject user. Organization field 1210 specifies an organization or company of which the subject user is a member or employee, respectively. It should be appreciated that user record 304 is merely illustrative and can include additional or different fields. It should also be appreciated that the fields of user record 304 can be user-customizable in some embodiments.

Account field 1212 references an account 1222 by which the subject user is granted access to package delivery system 100 (Figure 1). By conditioning processing of a package upon aspects of the account of the sender, the recipient can be mindful of sending large automated reply messages to a sender for whom package storage is a significant cost issue. Database identifier 1214 identifies the subject user within a directory 1224, thereby enabling the recipient to specify processing contingent on one or more attributes of the sender as stored in directory 1224.

Schedule pointer 1216 and user-defined state variables 1218 are described more completely below. Briefly, schedule pointer 1216 references a schedule data file 1226 which represents an appointment calendar or similar schedule for the subject user, and user-defined state variables 1218 store data representing aspects of the state of the subject user, e.g., on vacation, traveling on business, extended leave of absence, etc. Alternatively, the subject user's schedule can be represented by a scheduling system which is accessible through an API. In such an alternative embodiment, schedule pointer 1216 identifies the scheduling system and can include zero or more access parameters of the

scheduling system rather than a data file.

Package 306 (Figure 7) includes recipients field 704. Recipients are specified in a number of sub-fields, namely, TO sub-field 706, CC sub-field 708, and BCC sub-field 710 which specify, respectively, direct recipients, carbon-copied recipients, and blind carbon-copied recipients. However, in this illustrative embodiment, recipients specified in BCC sub-field 710 are not determinable by other recipients of package 306 and therefore cannot be the subject of any conditions specified by a recipient. Recipients — specified in either recipients field 704 itself or any sub-field thereof except BCC sub-field 710— can be included in conditions such as condition 602 (Figure 6) in a manner analogous to that described above with respect to sender field 702 (Figure 7). For example, recipients can be specified as matching a regular expression or by matching an attribute of a user record, e.g., user record 304 (Figure 3), corresponding to the recipient field or sub-field. In addition, actions, e.g., as represented by action record 604 (Figure 6), which send copies of a package do so in this illustrative embodiment by duplicating package 306 (Figure 7) and changing contents of TO sub-field 706 to specify the recipient to whom the copy is sent. In addition, the body of the message as represented in body field 714 described below can be modified to identify the copy as such. Testing for recipients is useful since the recipient can make certain actions conditional upon other recipients to whom the package is sent or upon whether the subject recipient is identified in TO sub-field 706 or CC sub-field 708. Conditions can apply to recipients in the manner described above with respect to senders including, for example, use of regular expressions and X.500 or similar directories.

Subject field 712 (Figure 7) of package 306 specifies a textual subject of the package for convenience in categorizing and handling of the package. Body field 714 of package record 206 stores the substantive content of the message of the package represented by package record 206. Conditions, e.g., condition record 602 (Figure 6), can match a subject or body using a boolean expression and/or a regular expression. Rules such as rule record 502 can use such conditions to detect terms or phrases in subject field 712 (Figure 7) and/or body field 714. which indicate priority (e.g., “urgent,” “important,” or “ASAP”), which indicate an unwanted commercial solicitation, or which are offensive to the recipient.

Body attributes field 716 of package 306 specifies characteristics of the body

represented in body field 714. Such attributes can include, for example, the particular format of the body, e.g., text, rich text format (RTF), or HTML, and the particular character set of which the body is composed. A condition involving body attributes can be used, for example, to detect packages with HTML bodies, and an associated action can convert the HTML body to a text or RTF body, thereby eliminating active components of HTML content such as javascript elements.

Delivery attributes 718 (Figure 7) specify the manner in which package 306 is delivered. For example, delivery attributes 718 can specify, for example, a relative priority of the package, whether a return receipt notification is requested by the sender, and a time at which to deliver the package.

Conditions such as condition record 602 (Figure 6) can specify specific delivery attributes. For example, packages with relatively high priority can cause the recipient to be notified, e.g., by an e-mail address directed to an alphanumeric pager or wireless telephone with text messaging capability such as SMS or WAP.

Actions of rules, e.g., action record 604, can modify the delivery attributes of a package. For example, a recipient can increase the relative priority of packages whose expiration approaches. Such a rule would be the equivalent of “if this package was sent by a customer and includes ‘ASAP’ in the subject or message body, increase the priority of this package.”

Custom attributes 720 can be used to specify characteristics of package 306 other than those specified in the other fields of package 306. In this illustrative embodiment, custom attributes 720 include a list of associated name/value pairs. In each pair, a name identifies the particular attribute and the value specifies the particular value of that attribute for package 306. Custom attributes 720 make package 306 extensible since attributes which are not conceived at the time system 100 is implemented can be added and represented in custom attributes 720.

Package 306 can also include keywords 722 which indicate significant portions of the content of package 306. Similar to subject field 712, keywords 722 provide a summation of the content of package 306 and can be used by the recipient to determine the nature of package 306 for appropriate processing.

Package 306 can include one or more attached data files. In particular, package

306 includes attached data file records 724 each of which references a respective data file which is considered attached to package 306. Attached data file 750 is such an attached data file. Alternatively, attached data files such as attached data file 750 are included within package 306 in an encoded form.

Attached data file 750 includes a name 752, a MIME (Multipurpose Internet Mail Extension) type 754, a size 756, custom attributes 758, and substantive content 760. Name 752 specifies a name of attached data file 750. MIME type 754 specifies a type of data file. For example, MIME type 754 can specify that attached data file 750 is a Microsoft Word document or a text document or a JPEG image. Size 756 specifies the size of attached data file 750. Custom attributes 758 represent subsequently defined attributes in a manner analogous to that described above with respect to custom attributes 720. For example, custom attributes 758 can include a number of attribute names and associated respective attribute values.

Conditions involving data file names as specified by name 752 can be used to detect specific files to detect packages which include data files of specific types. Conditions involving data file types as specified by MIME type 754 can be used to detect packages to which data files of specific types are attached. If a name of an attached data file does not accurately indicate the type of data file, MIME type 754 is likely to accurately indicate the type. While it is appreciated that the sender can change the name of a data file or convert the data file from one type to another to circumnavigate such rules, automatic execution of certain data files containing malicious computer instructions depends upon recognition of certain data file types by the operating system of the recipient. Accordingly, detection of data file types which can carry such malicious computer instructions is effective since attempts to defeat such detection will likely also defeat any automatic execution of the malicious computer instructions.

Conditions involving size 756 can be used to limit the size of attached data files of a package or the total size of a package. The recipient can limit the size of attached data files and/or the size of the entire package or can specify actions to be performed upon such a package having a certain total size or having attached data files of a certain size.

Conditions involving content 760 can examine the substantive content of attached data file 750. Such conditions can determine whether words and/or phrases which are

present in the substantive content of the attached data file indicate that the attached data file is an unwanted commercial solicitation or is offensive to the recipient or otherwise is of particular interest to the recipient. In addition, such conditions can scan the substantive content for malicious computer instructions such as Trojan horses or viruses.

Through either determining the type of attached data files or by examining the substantive content of attached data files, profile engine 214 (Figure 2) can determine whether an attached data file is executable, i.e., capable of execution within a computer. Profile engine 214 can similarly determine, by reference to the type or content of attached data files, whether an attached data file is capable of including macros — a collection of computer instructions embedded in otherwise non-executable data files. Conditions based on whether an attached data file is executable or is capable of including macros provide a useful means to determine the degree to which security is compromised by accepting a package including such an attached data file. Such attached data files can be scanned for malicious computer instructions such as viruses or Trojan horses and, if detected, can have those malicious computer instructions removed or such attached data files can be removed entirely from the package. If the recipient determines that security is so compromised that virus scanning is not to be relied upon, the entire attached data file can be removed or the entire package can be discarded. The sender can be sent an automatic reply message indicating that the malicious computer instructions were found, perhaps identifying the data file in which the malicious computer instructions were found. Similarly, the sender can be sent an automatic reply message indicating that executable data files or data files containing macros are not accepted by the recipient.

It should be noted that some data files include one or more embedded data files. For example, attached data file 750 can be an archive of one or more data files compressed in accordance with the known and ubiquitous ZIP compression format. Profile engine 214 therefore extracts embedded data files from any attached data files 750 and performs recipient-specified processing to each of the extracted files and extracts any embedded data files from the extracted data files in a recursive fashion. As a result, recipient-specified processing cannot be avoided by merely compressing an attached data file which would otherwise be singled out for processing in accordance with the profile established by the recipient.

Package-Independent Environmental Conditions

In addition to those conditions described above, rule 502 (Figure 6) can include conditions 602 which are independent of the contents or parameters of a particular package. One such condition is the time at which the package is to be delivered. Profile engine 214 (Figure 3) has access to a clock or time server within e-mail firewall 105.2 and can determine current date and time in a conventional manner. Accordingly, conditions can detect attempted delivery of a package at certain predetermined times such as non-business hours and weekends and holidays. For example, the recipient can have packages forwarded to a home e-mail address of the recipient if delivery of the package is attempted during non-business hours or days. By combining such a condition with a condition detecting high priority packages, the recipient can specify that only high priority packages are forwarded to the recipient's home e-mail address during such non-business hours and days, for example.

The recipient can also specify conditions related to time before or since an event. Such an event can be, for example, submission of a package, attempted delivery notification of the recipient, access of the package or one or more attached data files by the recipient, and expiration of the package. Example usages of such conditions include, for example, incrementing priority of the package and forwarding a copy of the package to a third party recipient if an amount of time has elapsed since arrival of the package at e-mail firewall 105.2 (Figure 1) without access to the package from the recipient (e.g., "forward this message to Joanne if I do not access this package within five days of its arrival"). In this illustrative example, e-mail firewall 105.2 also serves the function of receiving and holding packages for the recipient pending retrieval of such packages through e-mail network 103. In other words, e-mail firewall 105.2 also serves as the POP (Post Office Protocol) server for the recipient in this illustrative example. As a result, e-mail firewall 105.2 can determine (i) when a particular package arrives for the recipient and (ii) if and when the recipient retrieves the package to computer 114.

To test such a condition, profile engine 214 (Figure 2) determines a time at which evaluation of the condition should be performed and schedules such evaluation at that time. For example, if a condition is based upon five days after arrival of the package, profile engine 214 determines the time of the arrival of the package, adds five days to that

time to determine a condition time, and schedules a task which evaluates the condition to execute at the condition time. At the condition time, the scheduled task of the package processing module executes and evaluates the condition — by determining whether the package has been retrieved by the recipient for example.

As such time-based conditions are resolved at respective condition times, those conditions are replaced within a boolean expression such as boolean expression 900 (Figure 9) with the respective boolean values to which those conditions resolve. When no time-based conditions of a boolean expression remain unresolved, profile engine 214 (Figure 2) can determine the overall boolean value of the boolean expression to thereby determine whether to perform the associated actions of the rule, e.g., rule 502 (Figure 6).

Profile engine 214 (Figure 2) can also process a package conditionally upon the schedule of the recipient as represented in schedule data file 1224 (Figure 12). In this illustrative embodiment, e-mail firewall 105.2 (Figure 1) accesses a web-based calendar system similar to the web-based calendar system implemented by Netscape Communications Corporation and accessible through the World Wide Web at <http://calendar.netscape.com/>. Profile engine 214 (Figure 2) can query schedule data file 1224 to determine if the recipient is scheduled to be in a meeting and, if so, with whom. The recipient can specify, for example, that notification messages are sent to an alphanumeric pager of the recipient (i) if the recipient is in a meeting, (ii) only if the recipient is not in a meeting, or (iii) if the recipient is in a meeting and the package in question has a high priority or is from a specific sender. In addition, schedule data file 1224 can indicate holidays and processing can be conditioned upon whether a particular day is a holiday. As described above with respect to an alternative embodiment, the recipient's schedule can be specified in a scheduling system in which the recipient's schedule can be determined through an API.

As described briefly above, user-defined state variables 1218 (Figure 12) specify user-defined aspects of a user's state. Profile engine 214 (Figure 2) can use user-defined state variables 1218 (Figure 12) to condition processing of received packages in accordance with the recipient's state. For example, the recipient can specify that aspects of the recipient's state can include "on vacation" and "traveling on business." The recipient can thereafter toggle such aspects of the recipient's state on and off. For

example, the recipient can set an “on vacation” flag as the recipient begins a vacation. Accordingly, any processing of incoming packages is performed in the context of the recipient’s updated state, namely, in the context of the fact that the recipient is on vacation. When the recipient returns to work, the recipient can reset the “on vacation” flag to false to indicate that the recipient is no longer on vacation.

Profile Engine 214

To perform the functions described above, profile engine 214 interacts with profile managers 216 which include an access manager 218, a content manager 220, a format manager 222, a virus manager 224, and a security manager 226, each of which is all or part of one or more computer processes executing within one or more computers.

Access manager 218 is primarily responsible for controlling delivery of packages through SMTP relay 202. In particular, access manager 218 can delay or block delivery of packages through SMTP relay 202 in accordance with the profiles of the recipient. For example, the recipient can specify that packages from specified senders are not to be received or that packages exceeding a predetermined size are to be rejected. Access manager 218 blocks a package by discarding the package record, e.g., package 306, before SMTP relay 202 forwards the package record to the recipient. Access manager 218 can delay a package by placing the package in a queue for later delivery to the recipient. A package can be delayed for various purposes, including to allow others of profile managers 216 to process the package.

In one embodiment, various queues are established for packages of various levels of priority and packages of the higher queues are delivered before packages of lower priorities. Access manager 218 can increase or decrease the level of priority of a package and place the package in the appropriate queue for the new level of priority such that SMTP relay 202 delivers the package in accordance with its new level of priority.

Access manager 218 can also perform time-based processing of packages. Access manager 218 includes a clock mechanism or, alternatively, has access to a date and time server. Accordingly, access manager 218 is capable of evaluating time-based conditions in the manner described more completely above.

Content manager 220 performs content-based processing of packages in the

manner described more completely above. Content manager 220 evaluates conditions involving the content of subject field 712, body field 714, and content 760 (Figure 7) of attached data files 750, for example. In addition, any actions which require modification of the content of the package are carried out by content manager 220 (Figure 2).

Virus manager 224 analyzes attached data files 750 (Figure 7) for malicious computer instructions, e.g., viruses. In addition, virus manager 224 (Figure 2) determines whether attached data files 750 (Figure 7) are executable or can include macros in the manner described more completely above. In this illustrative embodiment, virus manager 224 can also detect malicious computer instructions within various types of compressed data files, including the known PKZip, PKLite, ARJ, LZExe, LHA, and MSCompress compressed data file formats. Virus manager 224 can use a conventional virus scanning engine. Virus scanning is known and conventional and is not described further herein. After an attached data file 750 (Figure 7) is scanned and determined to be free of known or detectable malicious computer instructions, virus manager 224 (Figure 2) marks the attached data file as clean by associating data so indicating with the attached data file 750 (Figure 7).

Format manager 222 converts packages, and/or attached data files, from one data format to another. For example, format manager 222 can convert a package from the known UUENCODE format to the known MIME format. In addition, format manager 222 can convert attached data files 750 (Figure 7) from one data format to another using conventional techniques. In one embodiment, others of profile managers 216 process packages in the MIME format, and format manager 222 converts the package to the MIME format prior to processing by others of profile managers 216.

Security manager 226 handles encryption and decryption of package 306 or parts thereof. For example, if e-mail network 103 is private such that package 306 can be securely delivered to the recipient therethrough and package 306 is encrypted, security manager 226 can decrypt package 306 including any attached data files for delivery to the recipient. Such decryption enables implementation of the recipient's profile upon packages which are encrypted. To enable decryption on behalf of the recipient, configuration 230 includes a public/private key pair 1220 (Figure 12) of the recipient in the recipient's user record 304. Security manager 226 (Figure 2) interacts with an

S/MIME module 215 to decrypt and encrypt packages according to the known S/MIME secure e-mail protocol.

As described above, configuration 230 stores profile records, e.g., profile record 308 (Figure 3), and associates the profile records with user records, e.g., user record 304, to which the profile records pertain. In this illustrative embodiment, each user record includes a reference to a list of all profile records which pertain to the user, and each profile record includes a reference to the one or more user records representing the users to which the profile record pertains.

The profile records stored by configuration 230 (Figure 2) can be in any format convenient for profile engine 214 and profile manager 216. For example, profile records can represent profiles in any of the textual formats described below or in a binary representation in which similar information is stored. Profile records can be stored as one or more flat data files, as a relational database, or as an object oriented database. Flat data files, relational databases, and object oriented databases are known and are described further herein.

Profile managers 216 collectively include logic which implements profiles of a recipient in the manner described above with respect to logic flow diagram 400 (Figure 4).

Configuration 230 (Figure 2) interacts with the recipient through computer system 104 (Figure 1) and e-mail network 103 to define one or more profiles, e.g., profile record 308 (Figure 3), which are applicable to packages sent to the recipient. Configuration 230 (Figure 2) can interact with the recipient in any of a number of ways. For example, the recipient can submit a textual data file specifying a profile and configuration 230 can parse the textual data file, form a profile record such as profile record 308 (Figure 3) and submit the profile record to configuration 230 (Figure 2) for storage in datastore 120. Possible textual formats for profiles are described more completely below.

Alternatively, configuration 230 (Figure 2) can provide an interactive interface by which the recipient can add, delete, and modify rules of a profile. Similarly, the interface provides mechanisms by which the recipient can add, delete, and modify conditions and actions of a specific rule when adding or modifying a rule of the profile. In specifying — by addition or modification — a condition, the recipient is prompted for (i) a parameter of

a package or a package-independent parameter, (ii) a relation, and (iii) a data value. Parameters include, for example, those described above with respect to package 306 in Figure 7 and those package-independent parameters described above, and the recipient can be presented with a list of such parameters from which to select a parameter. Relations can include such relations as “contains,” “is equal to,” “is greater than,” “is less than,” and negations of each such relation, and the recipient can select such a relation from a list of available relations. The recipient specifies a data value by entering the value. Configuration 230 (Figure 2) in this illustrative embodiment verifies that the entered data value conforms to any validity constraints imposed upon the selected package parameter. For example, if the selected package parameter of the condition is a date, configuration 230 ensures that the condition data value entered by the recipient is a valid date in the same manner that the package parameter is verified to be a valid date.

In one embodiment, the interactive interface of configuration 230 is implemented as a set of one or more HTML forms. HTML forms are known and are not described further herein. In an alternative embodiment, the interactive interface is implemented by all or part of one or more computer processes which convert the profiles specified by the recipient to one or more data files representing the profiles in a format which is recognized by configuration 230 (Figure 2). For example, the format can be any of the textual formats described below.

If configuration 230 recognizes profiles in a standard format, such as a textual format; conventional editors executing within computer 114 (Figure 1) can be used by the recipient to specify a profile which is submitted through e-mail network 103 to server 108 to configuration 230 (Figure 2). For example, the NOTEPAD and WORDPAD programs available from Microsoft Corporation of Redmond, Washington in conjunction with their WINDOWS® family of operating systems can be used to edit profiles in the textual formats described below.

Two illustrative formats for profile specification are described herein: a rule list and a scripting language. Each can be represented textually, e.g., in the known ASCII and XML formats, or as binary data.

The rule list format is a simple list of rules, each of which is a pairing or association of a list of one or more conditions with a list of one or more actions. The

following grammar illustrates the rule list format:

Profile = list of Rule

Rule = Conditions Actions

Conditions = a boolean expression using zero or more
instances of Condition

Actions = list of Action

Condition = DeliveryAttribute or ExternalAttribute

DeliveryAttribute = PackageAttribute OR SenderAttribute
OR RecipientAttribute OR EventAttribute

ExternalAttribute = CurrentTime OR CurrentDate OR
RandomNumber OR

CustomAttribute(attributeName) OR etc.

PackageAttribute = Subject OR Body OR TimeOfDelivery
OR DeliverySecurityAttribute OR
CustomAttribute(AttributeName) OR list of
FileAttribute

FileAttribute = FileName OR MIMEType OR FileSize OR
FileTextualContent OR
CustomAttribute(AttributeName) OR list of
FileAttribute

SenderAttribute = SenderEmailAddress OR
SenderAttributeFromDirectoryLookup(AttributeName)
OR CustomAttribute(AttributeName)

RecipientAttribute = RecipientEmailAddress OR
RecipientAttributeFromDirectoryLookup(AttributeName)
OR CustomAttribute(AttributeName) OR
ScheduleLookup(date, time) OR
RecipientStateVariable(VariableName)

EventAttribute = TimeBefore(Event) OR TimeSince(Event)

Event = SubmissionOfPackage OR ExpirationOfPackage OR
DeliveryNotification OR PackageRetrieval OR etc.

Action = Discard OR AutoReply(ReplyPackage) OR
 Redirect(recipient) OR SendCopyTo(recipient) OR
 Notify(recipient) OR RemoveAllAttachments OR
 RemoveAttachmentsMatchingCondition OR
 AppendToBody OR PrependToBody OR
 ModifyDeliveryOption(Option, NewValue) OR
 ConvertAttachmentFormat(NewFormat) OR
 CompressAttachment OR
 RunProgramForAttachment(ProgramName) OR
 CleanVirusFromAttachment OR etc.

FileAttribute has a recursive definition since some file formats include a list of embedded files. For example, compressed data formats such as the popular and known ZIP compressed data format embeds a number of files within a compressed file. In addition, each embedded file can also have embedded files, e.g., can be a compressed data file in the ZIP format.

The scripting language format represents processing performed on behalf of the recipient in the form of a scripting language. In one embodiment, a number of predefined objects express conditions in the known ECMA-262 scripting language of the European Computer Manufacturers Association (ECMA). ECMA-22 (sometimes referred to as ECMAScript or JavaScript) is known and is not described further herein. In this embodiment, actions are represented by predefined methods in the ECMA-262 scripting language.

The following objects are illustrative examples of objects which can represent parameters: sender.directoryLookup(directory, attributeName), recipient.directoryLookup(directory, attributeName), package.subject, package.body, package.sendDate, package.deliveryTime, package.priority, package.securityOption, package.file[index], package.file[index].length, package.file[index].name, package.file[index].mimeType, package.file[index].hasVirus(), package.file[index].isExecutable(), package.file[index].hasMacros(), package.file.scanText("regular expression"), current.time, current.date, timeSince(event), timeBefore(event), recipient.schedule(time, date), and customAttribute(name).

The following methods are illustrative examples of methods which can represent actions: `package.discard()`, `package.AutoReply("I am on vacation but will be back in the office on Tuesday.")`, `package.redirect("e-mail address")`, `package.sendCopyTo("e-mail address")`, `package.body.append("This message is privileged as Attorney/Client communication.")`, `package.files.removeAt(index)`, and `package.submitForProcessing("Program")`. In addition, actions can be represented as object properties which can be written in the scripting language. For example, "(URGENT)" can be appended to the subject by the script instruction:

```
package.subject += "(URGENT)"
```

Similarly, a copy of the package can be sent to an alternative e-mail address by the script instruction:

```
package.CC += "pager@myisp.com"
```

The rules list format and script format can be combined. For example, conditions can be expressed in the rules list format while actions are expressed as scripts. Alternatively, conditions can be expressed as scripts while actions are expressed in the rules list format described above. Furthermore, these illustrative formats are exactly that: illustrative. Other formats are possible for specifying conditions and associated actions to be taken if the conditions are satisfied.

The above description is illustrative only and is not limiting. For example, it should be appreciated that processing of a package can be automatically processed on behalf of the sender and/or a policy authority of the sender in a manner described in International Patent Application PCT/US98/15552 for which the United States is a designated state and which claims priority of U.S. Patent Application S/N 60/053,668, and that description is incorporated herein by reference. In this illustrative embodiment, any automated processing on behalf of the sender is performed first, followed by processing in

a manner specified by a policy authority of the sender, and then followed by automated processing on behalf of the recipient. Specifically, the present invention is defined solely by the claims which follow and their full range of equivalents.

What is claimed is:

1. A method for processing a message to be delivered from a sender to a recipient, the method comprising:
receiving the message from the sender;
processing the message in accordance with logic associated with the recipient; and
upon a condition in which the processing of the message indicates that the message is to be delivered to the recipient, sending the message to the recipient at a remotely located computer.
2. The method of Claim 1 wherein processing comprises:
determining that the message satisfies one or more conditions; and
if the message satisfies the one or more conditions, performing one or more actions which are associated with the one or more conditions.
3. The method of Claim 2 wherein the one or more actions discard the message.
4. The method of Claim 2 wherein the one or more actions redirect the message to another recipient.
5. The method of Claim 2 wherein the one or more actions send a predetermined reply message to the sender.
6. The method of Claim 2 wherein the one or more actions send a predetermined notification message to a predetermined notification recipient.
7. The method of Claim 6 wherein the notification recipient is an alternative

address of the recipient.

8. The method of Claim 7 wherein the alternative address of the recipient directs the notification message to an alphanumeric pager of the recipient.

9. The method of Claim 7 wherein the alternative address of the recipient directs the notification message to a text-messaging-capable telephone of the recipient.

10. The method of Claim 9 wherein the text-messaging-capable telephone receives messages according to a short message service.

11. The method of Claim 9 wherein the text-messaging-capable telephone receives messages according to a wireless application protocol.

12. The method of Claim 6 wherein the notification message includes a subject of the first-mentioned message.

13. The method of Claim 6 wherein the notification message includes at least a part of a message body of the first-mentioned message.

14. The method of Claim 2 wherein the one or more actions send a copy of the message to another recipient.

15. The method of Claim 2 wherein the one or more actions initiate execution of a computer program to process the package.

16. The method of Claim 2 wherein the one or more conditions include a boolean expression involving data related to the sender.

17. The method of Claim 2 wherein the one or more conditions include a boolean expression involving data related to one or more of the recipients.

18. The method of Claim 2 wherein the one or more conditions include a boolean expression involving data related to one or more attributes of the message.
19. The method of Claim 1 wherein the profile data is received from the recipient through the second computer network.
20. The method of Claim 19 wherein the computer network is an intranet.
21. The method of Claim 1 wherein the message is also addressed to one or more other recipients.
22. The method of Claim 1 wherein the message includes one or more data files.
23. The method of Claim 1 wherein receiving is according to a post office protocol.
24. The method of Claim 1 wherein the sending is according to a simple mail transport protocol.
25. The method of Claim 1 further comprising:
 - retrieving data representing one or more characteristics of the sender from a database according to data identifying the sender;
 - wherein the logic includes one or more conditions based upon the one or more characteristics.
26. The method of Claim 25 wherein the database is accessed according to a lightweight directory access protocol.
27. The method of Claim 25 wherein the database is accessed according to an

X.500 directory access protocol.

28. The method of Claim 1 further comprising:
retrieving data representing one or more characteristics of a selected one of
the other recipients from a database according to data identifying the sender;
wherein the logic includes one or more conditions based upon the one or
more characteristics.

29. The method of Claim 28 wherein the database is accessed according to a
lightweight directory access protocol.

30. The method of Claim 28 wherein the database is accessed according to an
X.500 directory access protocol.

31. The method of Claim 1 further comprising:
determining that one or more delivery security parameters are specified in
the data package;
wherein the logic includes one or more conditions based upon the one or
more delivery security parameters.

32. The method of Claim 1 further comprising:
determining that delivery of the data package is subject to a delivery
schedule;
wherein the logic includes one or more conditions based upon the delivery
schedule.

33. The method of Claim 1 wherein the logic includes one or more conditions
based upon one or more times relative to one or more events.

34. The method of Claim 33 wherein the one or more relative times include a
time before an associated event.

35. The method of Claim 33 wherein the one or more relative times include a time since an associated event.

36. The method of Claim 33 wherein the one or more events include submission of the data package for delivery to the selected recipient.

37. The method of Claim 33 wherein the one or more events include attempted delivery of the data package to the selected recipient.

38. The method of Claim 33 wherein the one or more events include sending of notification of the data package to the selected recipient.

39. The method of Claim 33 wherein the one or more events include expiration of the data package.

40. The method of Claim 1 wherein the logic includes one or more conditions based upon one or more elements of a state defined by the selected recipient.

41. The method of Claim 40 further comprising:
receiving state data from the selected recipient wherein the state data represents at least one of the elements of the state; and
setting the state in accordance with the state data such that subsequently received messages addressed to the selected recipient are processed in accordance with the state so set.

42. The method of Claim 1 wherein the logic includes one or more conditions based upon a schedule of the selected recipient.

43. The method of Claim 42 wherein processing comprises:
determining a time associated with the data package; and
retrieving data representing the schedule of the selected recipient at the time

from a calendar system.

44. A computer readable medium useful in association with a computer which includes a processor and a memory, the computer readable medium including computer instructions which are configured to cause the computer to process a message to be delivered from a sender to a recipient by:

receiving the message from the sender;

processing the message in accordance with logic associated with the recipient; and

upon a condition in which the processing of the message indicates that the message is to be delivered to the recipient, sending the message to the recipient at a remotely located computer.

45. The computer readable medium of Claim 44 wherein processing comprises:
determining that the message satisfies one or more conditions; and
if the message satisfies the one or more conditions, performing one or more actions which are associated with the one or more conditions.

46. The computer readable medium of Claim 45 wherein the one or more actions discard the message.

47. The computer readable medium of Claim 45 wherein the one or more actions redirect the message to another recipient.

48. The computer readable medium of Claim 45 wherein the one or more actions send a predetermined reply message to the sender.

49. The computer readable medium of Claim 45 wherein the one or more actions send a predetermined notification message to a predetermined notification recipient.

50. The computer readable medium of Claim 49 wherein the notification recipient is an alternative address of the recipient.

51. The computer readable medium of Claim 50 wherein the alternative address of the recipient directs the notification message to an alphanumeric pager of the recipient.

52. The computer readable medium of Claim 50 wherein the alternative address of the recipient directs the notification message to a text-messaging-capable telephone of the recipient.

53. The computer readable medium of Claim 52 wherein the text-messaging-capable telephone receives messages according to a short message service.

54. The computer readable medium of Claim 52 wherein the text-messaging-capable telephone receives messages according to a wireless application protocol.

55. The computer readable medium of Claim 49 wherein the notification message includes a subject of the first-mentioned message.

56. The computer readable medium of Claim 49 wherein the notification message includes at least a part of a message body of the first-mentioned message.

57. The computer readable medium of Claim 45 wherein the one or more actions send a copy of the message to another recipient.

58. The computer readable medium of Claim 45 wherein the one or more actions initiate execution of a computer program to process the package.

59. The computer readable medium of Claim 45 wherein the one or more conditions include a boolean expression involving data related to the sender.

60. The computer readable medium of Claim 45 wherein the one or more conditions include a boolean expression involving data related to one or more of the recipients.

61. The computer readable medium of Claim 45 wherein the one or more conditions include a boolean expression involving data related to one or more attributes of the message.

62. The computer readable medium of Claim 44 wherein the profile data is received from the recipient through the second computer network.

63. The computer readable medium of Claim 62 wherein the computer network is an intranet.

64. The computer readable medium of Claim 44 wherein the message is also addressed to one or more other recipients.

65. The computer readable medium of Claim 44 wherein the message includes one or more data files.

66. The computer readable medium of Claim 44 wherein receiving is according to a post office protocol.

67. The computer readable medium of Claim 44 wherein the sending is according to a simple mail transport protocol.

68. The computer readable medium of Claim 44 wherein the computer instructions are configured to cause the computer to process the message by also:
retrieving data representing one or more characteristics of the sender from a database according to data identifying the sender;
wherein the logic includes one or more conditions based upon the one or

more characteristics.

69. The computer readable medium of Claim 68 wherein the database is accessed according to a lightweight directory access protocol.

70. The computer readable medium of Claim 68 wherein the database is accessed according to an X.500 directory access protocol.

71. The computer readable medium of Claim 44 wherein the computer instructions are configured to cause the computer to process the message by also:
retrieving data representing one or more characteristics of a selected one of the other recipients from a database according to data identifying the sender;
wherein the logic includes one or more conditions based upon the one or more characteristics.

72. The computer readable medium of Claim 71 wherein the database is accessed according to a lightweight directory access protocol.

73. The computer readable medium of Claim 71 wherein the database is accessed according to an X.500 directory access protocol.

74. The computer readable medium of Claim 44 wherein the computer instructions are configured to cause the computer to process the message by also:
determining that one or more delivery security parameters are specified in the data package;
wherein the logic includes one or more conditions based upon the one or more delivery security parameters.

75. The computer readable medium of Claim 44 wherein the computer instructions are configured to cause the computer to process the message by also:
determining that delivery of the data package is subject to a delivery

schedule;

wherein the logic includes one or more conditions based upon the delivery schedule.

76. The computer readable medium of Claim 44 wherein the logic includes one or more conditions based upon one or more times relative to one or more events.

77. The computer readable medium of Claim 76 wherein the one or more relative times include a time before an associated event.

78. The computer readable medium of Claim 76 wherein the one or more relative times include a time since an associated event.

79. The computer readable medium of Claim 76 wherein the one or more events include submission of the data package for delivery to the selected recipient.

80. The computer readable medium of Claim 76 wherein the one or more events include attempted delivery of the data package to the selected recipient.

81. The computer readable medium of Claim 76 wherein the one or more events include sending of notification of the data package to the selected recipient.

82. The computer readable medium of Claim 76 wherein the one or more events include expiration of the data package.

83. The computer readable medium of Claim 44 wherein the logic includes one or more conditions based upon one or more elements of a state defined by the selected recipient.

84. The computer readable medium of Claim 83 wherein the computer instructions are configured to cause the computer to process the message by also:

receiving state data from the selected recipient wherein the state data represents at least one of the elements of the state; and

setting the state in accordance with the state data such that subsequently received messages addressed to the selected recipient are processed in accordance with the state so set.

85. The computer readable medium of Claim 44 wherein the logic includes one or more conditions based upon a schedule of the selected recipient.

86. The computer readable medium of Claim 85 wherein processing comprises:
determining a time associated with the data package; and
retrieving data representing the schedule of the selected recipient at the time from a calendar system.

87. A computer system comprising:
a processor;
a memory operatively coupled to the processor; and
a message processing module (i) which executes in the processor from the memory and (ii) which, when executed by the processor, causes the computer to process a message to be delivered from a sender to a recipient by:
receiving the message from the sender;
processing the message in accordance with logic associated with the recipient; and
upon a condition in which the processing of the message indicates that the message is to be delivered to the recipient, sending the message to the recipient at a remotely located computer.

88. The computer system of Claim 87 wherein processing comprises:
determining that the message satisfies one or more conditions; and
if the message satisfies the one or more conditions, performing one or more actions which are associated with the one or more conditions.

89. The computer system of Claim 88 wherein the one or more actions discard the message.

90. The computer system of Claim 88 wherein the one or more actions redirect the message to another recipient.

91. The computer system of Claim 88 wherein the one or more actions send a predetermined reply message to the sender.

92. The computer system of Claim 88 wherein the one or more actions send a predetermined notification message to a predetermined notification recipient.

93. The computer system of Claim 92 wherein the notification recipient is an alternative address of the recipient.

94. The computer system of Claim 93 wherein the alternative address of the recipient directs the notification message to an alphanumeric pager of the recipient.

95. The computer system of Claim 93 wherein the alternative address of the recipient directs the notification message to a text-messaging-capable telephone of the recipient.

96. The computer system of Claim 95 wherein the text-messaging-capable telephone receives messages according to a short message service.

97. The computer system of Claim 95 wherein the text-messaging-capable telephone receives messages according to a wireless application protocol.

98. The computer system of Claim 92 wherein the notification message includes a subject of the first-mentioned message.

99. The computer system of Claim 92 wherein the notification message includes at least a part of a message body of the first-mentioned message.
100. The computer system of Claim 88 wherein the one or more actions send a copy of the message to another recipient.
101. The computer system of Claim 88 wherein the one or more actions initiate execution of a computer program to process the package.
102. The computer system of Claim 88 wherein the one or more conditions include a boolean expression involving data related to the sender.
103. The computer system of Claim 88 wherein the one or more conditions include a boolean expression involving data related to one or more of the recipients.
104. The computer system of Claim 88 wherein the one or more conditions include a boolean expression involving data related to one or more attributes of the message.
105. The computer system of Claim 87 wherein the profile data is received from the recipient through the second computer network.
106. The computer system of Claim 105 wherein the computer network is an intranet.
107. The computer system of Claim 87 wherein the message is also addressed to one or more other recipients.
108. The computer system of Claim 87 wherein the message includes one or more data files.

109. The computer system of Claim 87 wherein receiving is according to a post office protocol.

110. The computer system of Claim 87 wherein the sending is according to a simple mail transport protocol.

111. The computer system of Claim 87 wherein the message processing module causes the computer to process the message by also:

retrieving data representing one or more characteristics of the sender from a database according to data identifying the sender;

wherein the logic includes one or more conditions based upon the one or more characteristics.

112. The computer system of Claim 111 wherein the database is accessed according to a lightweight directory access protocol.

113. The computer system of Claim 111 wherein the database is accessed according to an X.500 directory access protocol.

114. The computer system of Claim 87 wherein the message processing module causes the computer to process the message by also:

retrieving data representing one or more characteristics of a selected one of the other recipients from a database according to data identifying the sender;

wherein the logic includes one or more conditions based upon the one or more characteristics.

115. The computer system of Claim 114 wherein the database is accessed according to a lightweight directory access protocol.

116. The computer system of Claim 114 wherein the database is accessed according to an X.500 directory access protocol.

117. The computer system of Claim 87 wherein the message processing module causes the computer to process the message by also:

determining that one or more delivery security parameters are specified in the data package;

wherein the logic includes one or more conditions based upon the one or more delivery security parameters.

118. The computer system of Claim 87 wherein the message processing module causes the computer to process the message by also:

determining that delivery of the data package is subject to a delivery schedule;

wherein the logic includes one or more conditions based upon the delivery schedule.

119. The computer system of Claim 87 wherein the logic includes one or more conditions based upon one or more times relative to one or more events.

120. The computer system of Claim 119 wherein the one or more relative times include a time before an associated event.

121. The computer system of Claim 119 wherein the one or more relative times include a time since an associated event.

122. The computer system of Claim 119 wherein the one or more events include submission of the data package for delivery to the selected recipient.

123. The computer system of Claim 119 wherein the one or more events include attempted delivery of the data package to the selected recipient.

124. The computer system of Claim 119 wherein the one or more events include

sending of notification of the data package to the selected recipient.

125. The computer system of Claim 119 wherein the one or more events include expiration of the data package.

126. The computer system of Claim 87 wherein the logic includes one or more conditions based upon one or more elements of a state defined by the selected recipient.

127. The computer system of Claim 126 wherein the message processing module causes the computer to process the message by also:

receiving state data from the selected recipient wherein the state data represents at least one of the elements of the state; and

setting the state in accordance with the state data such that subsequently received messages addressed to the selected recipient are processed in accordance with the state so set.

128. The computer system of Claim 87 wherein the logic includes one or more conditions based upon a schedule of the selected recipient.

129. The computer system of Claim 128 wherein processing comprises:

determining a time associated with the data package; and

retrieving data representing the schedule of the selected recipient at the time from a calendar system.

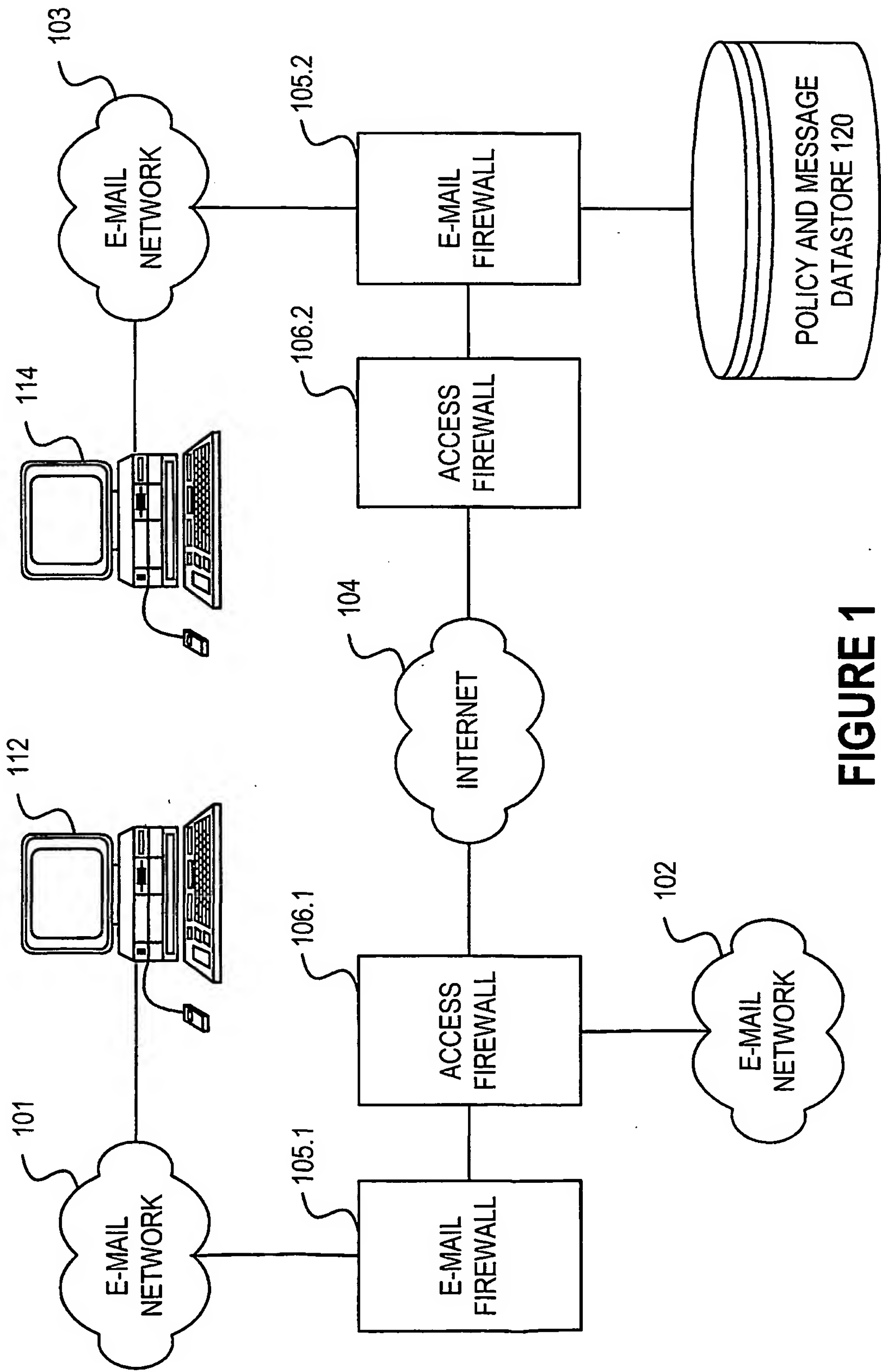
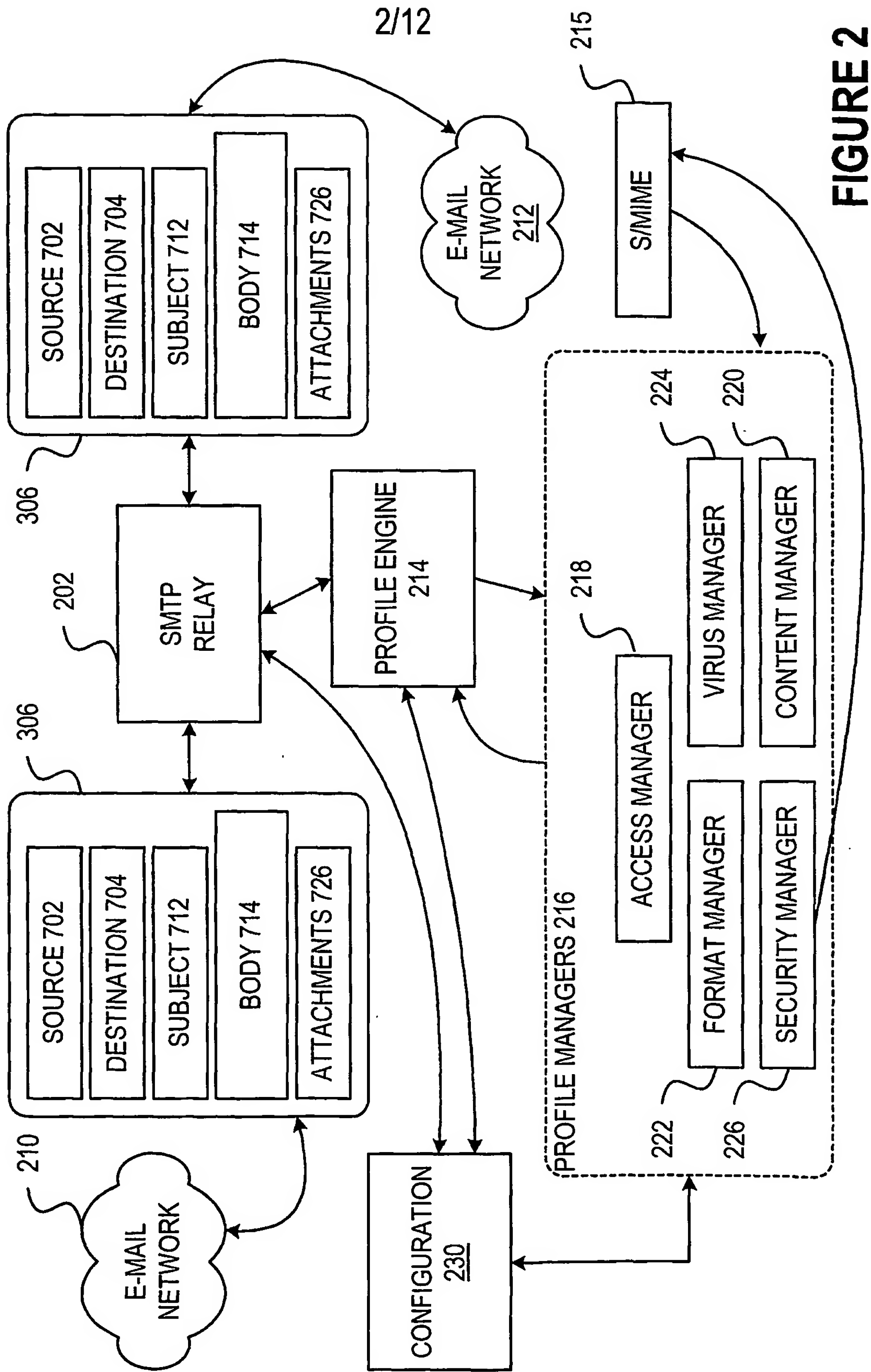


FIGURE 1



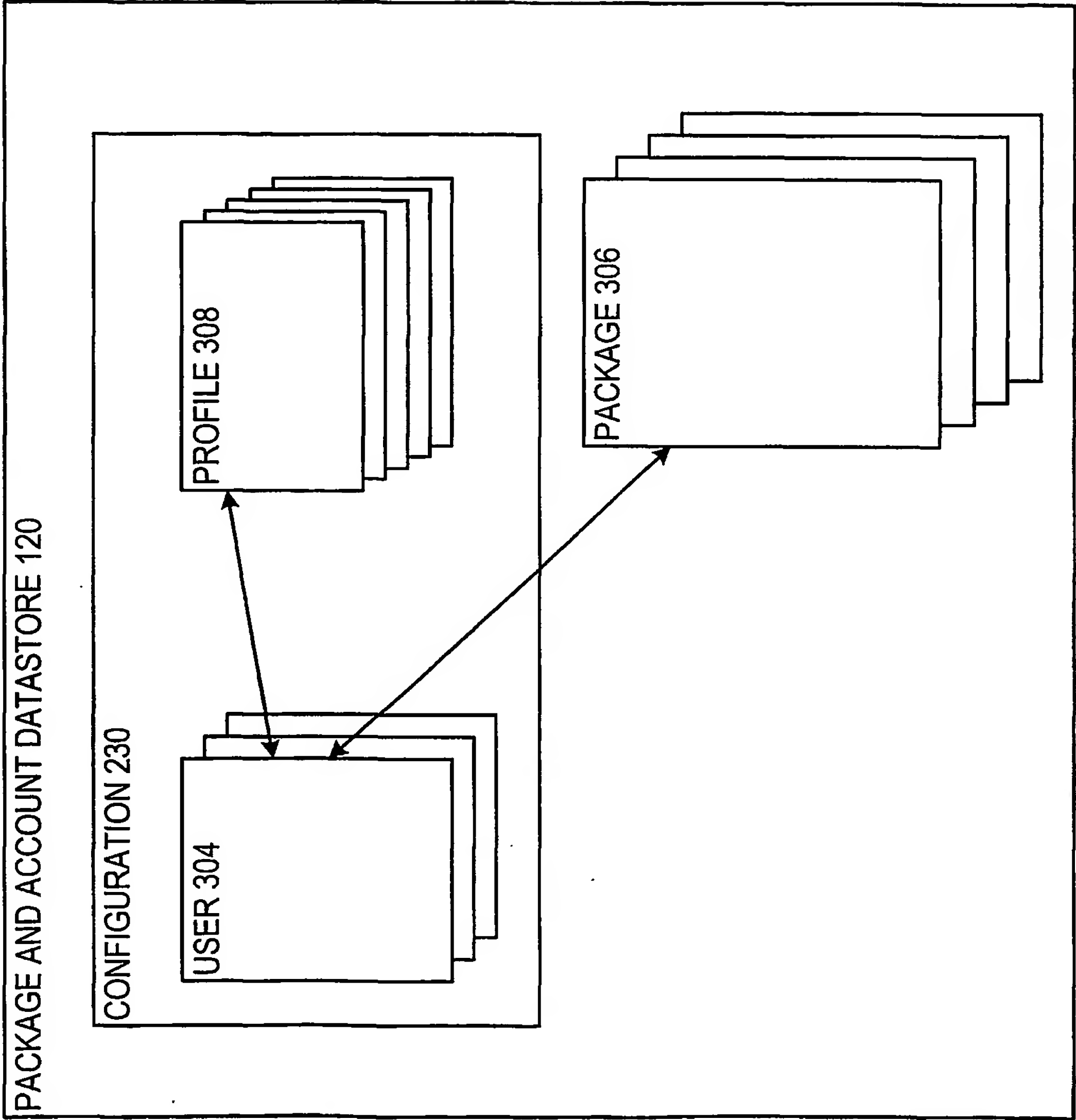


FIGURE 3

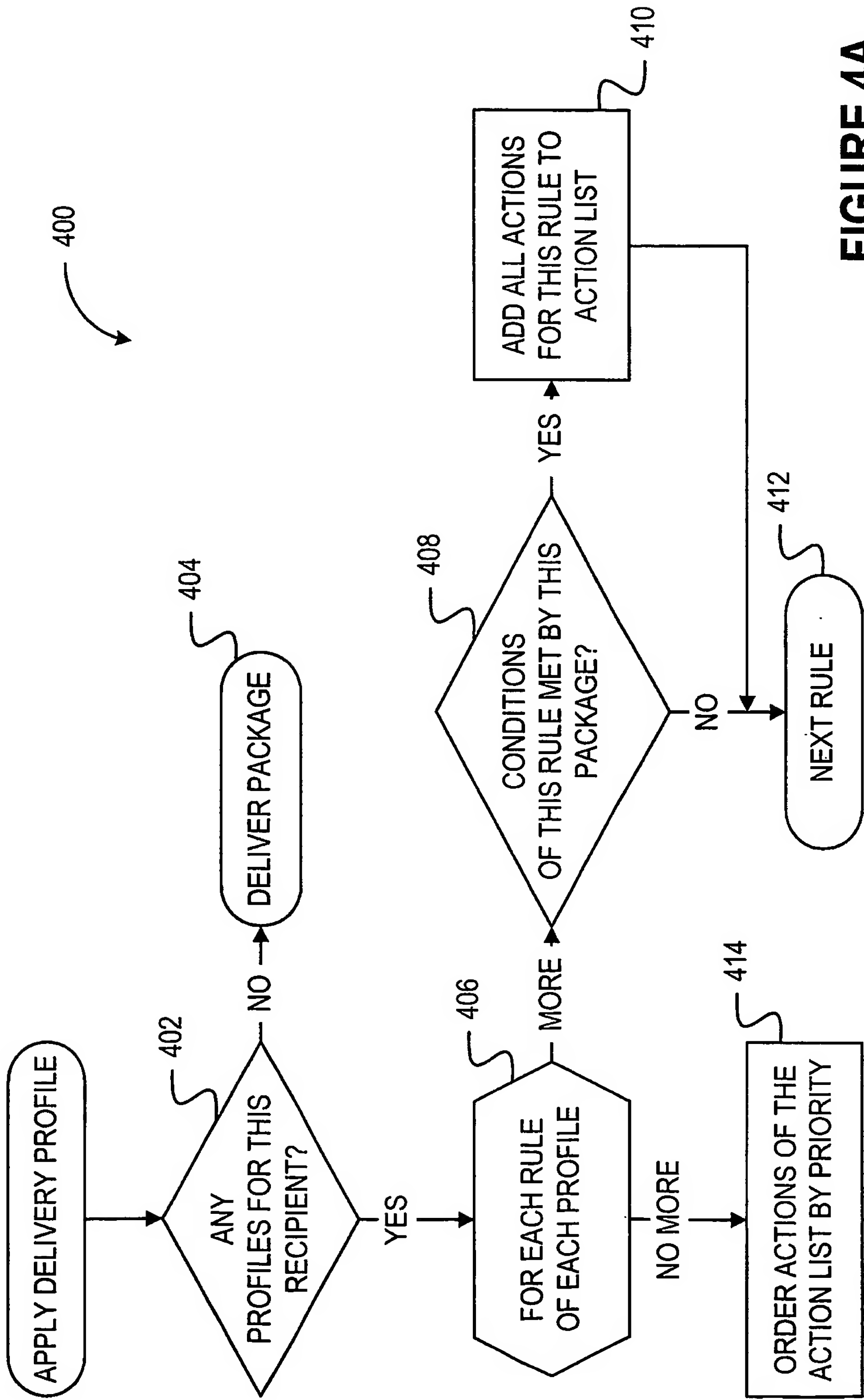


FIGURE 4A

FIGURE 4B

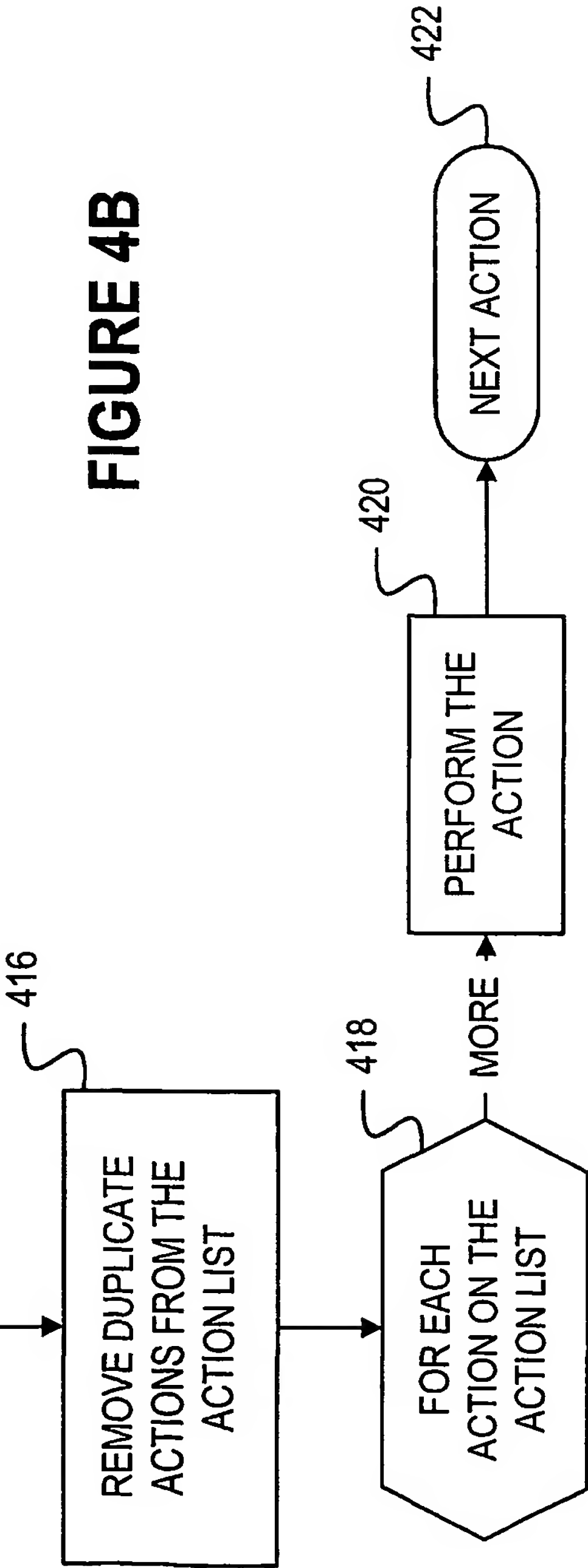


FIGURE 4A

FIGURE 4B

FIGURE 4

6/12

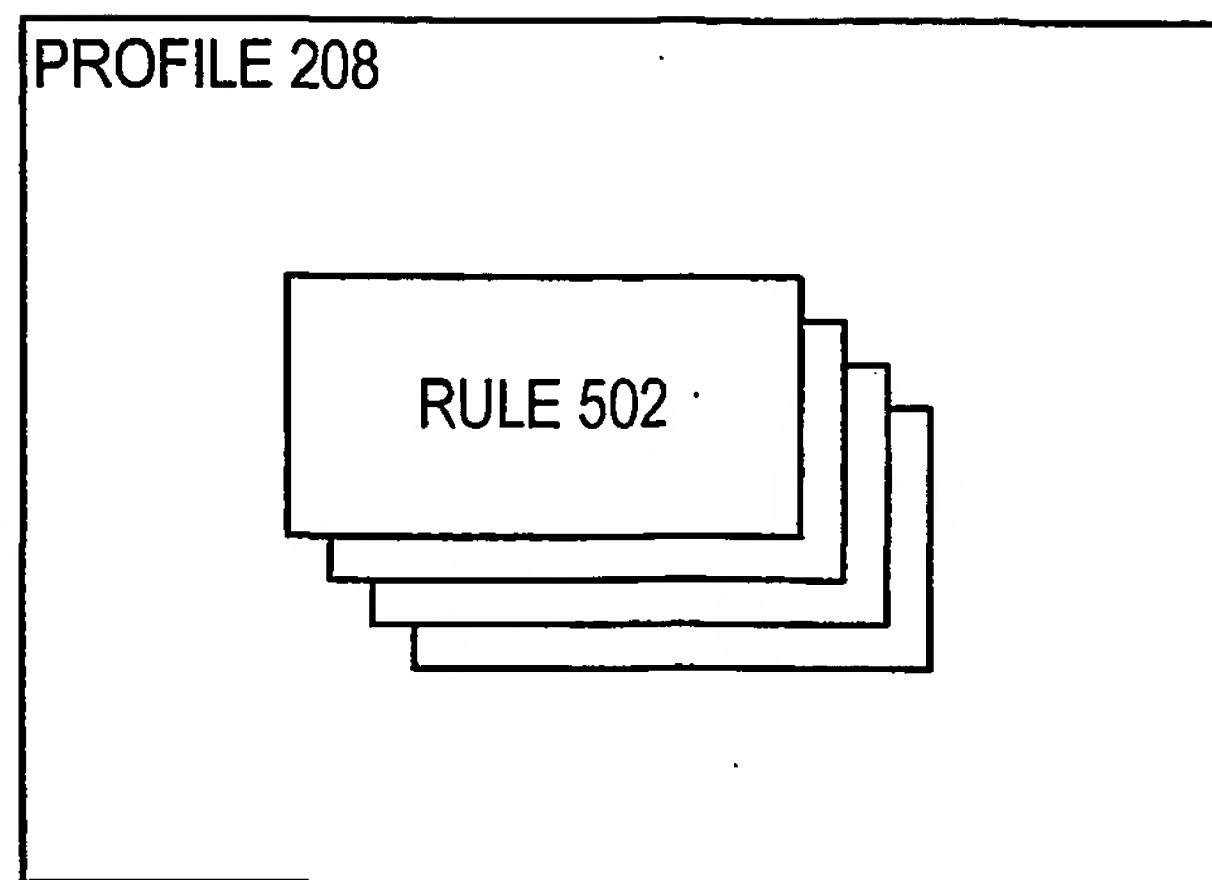


FIGURE 5

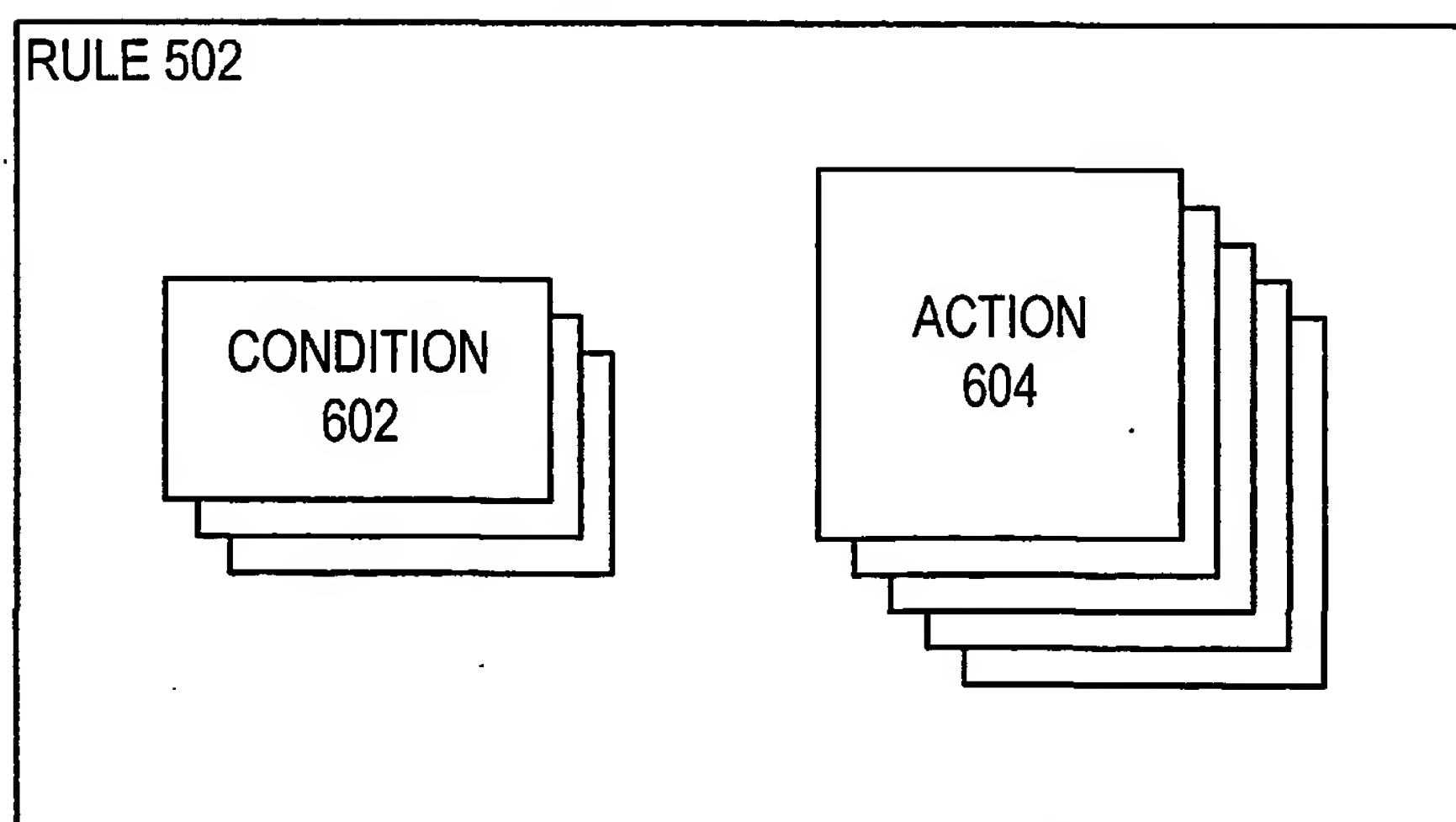
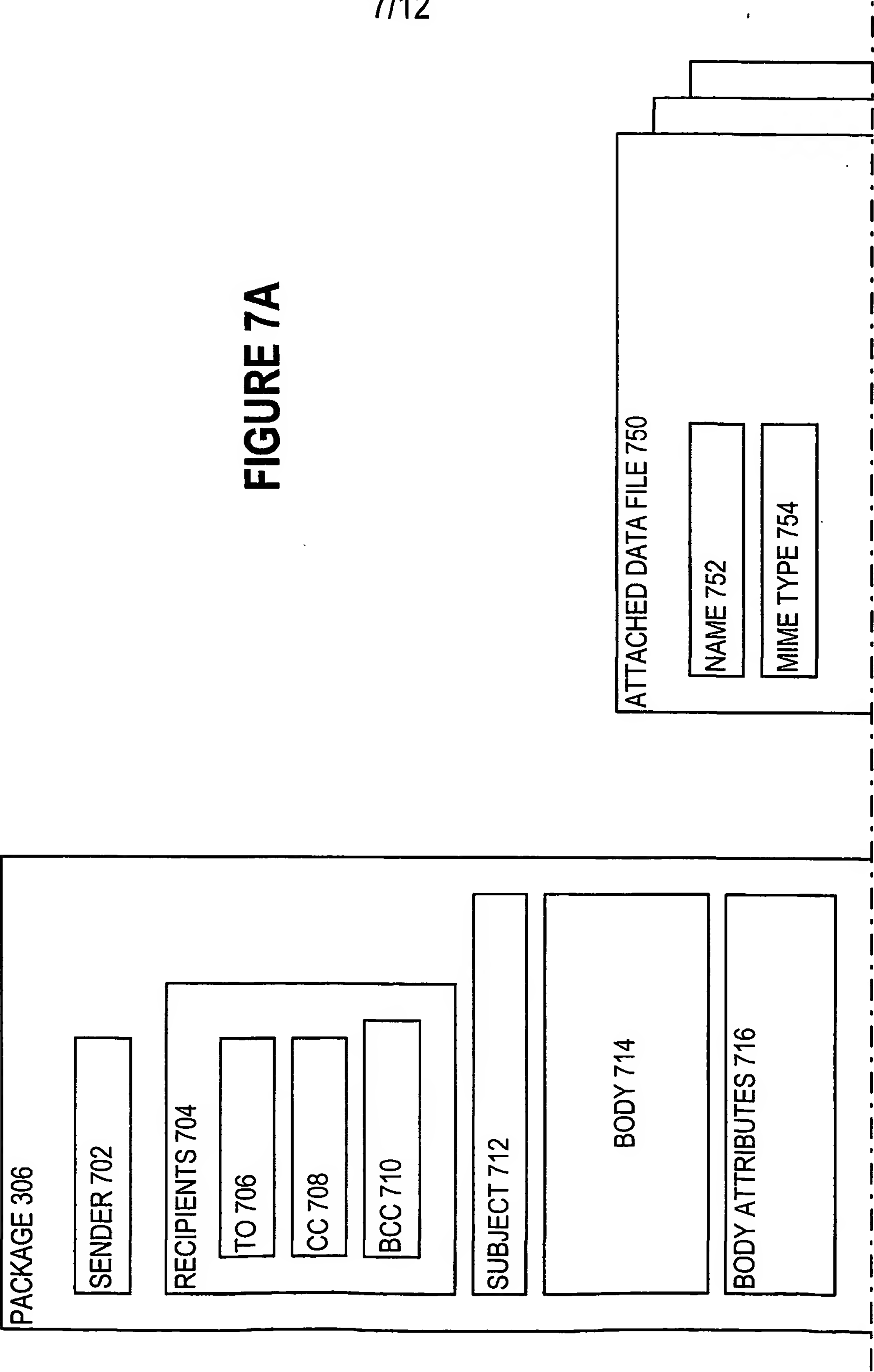


FIGURE 6

FIGURE 7A



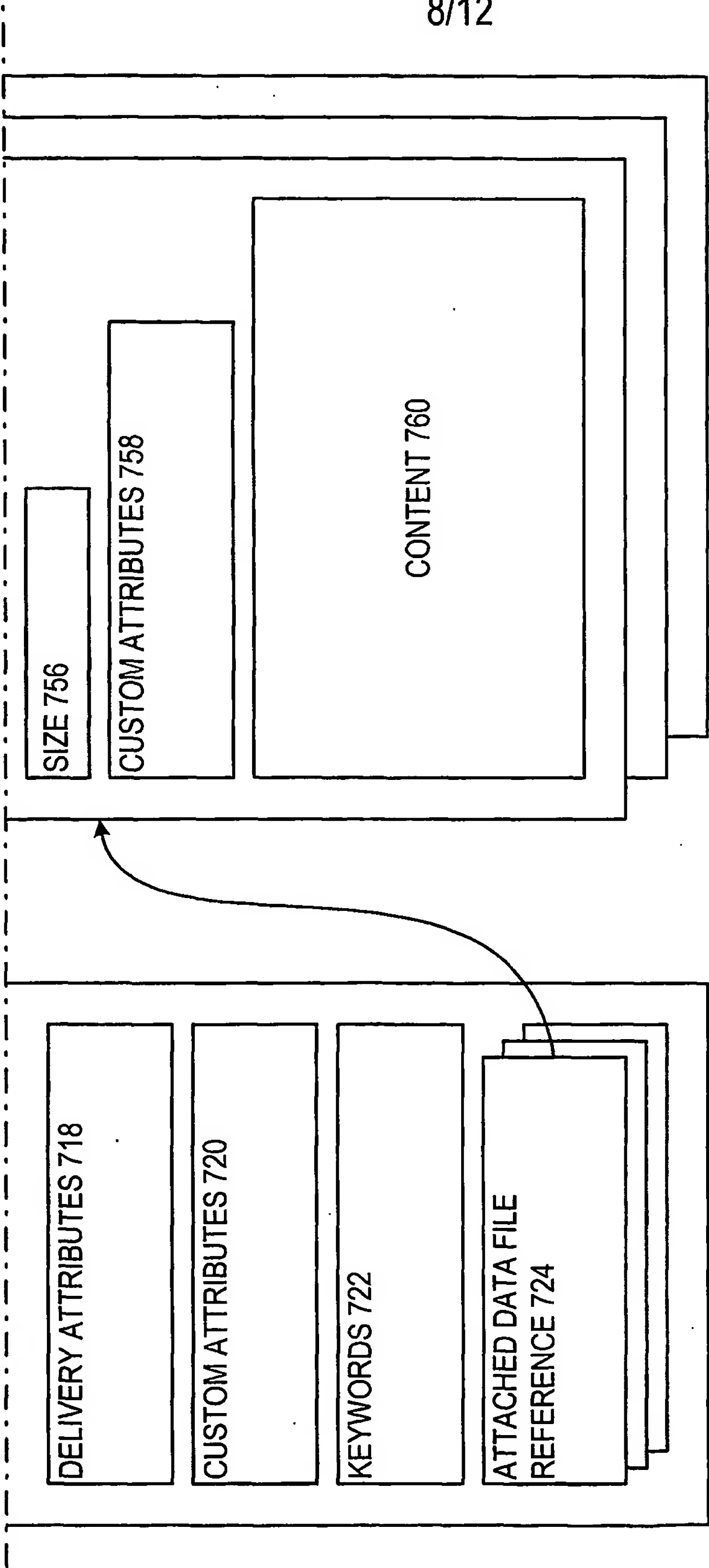


FIGURE 7B

FIGURE 7A

FIGURE 7B

FIGURE 7

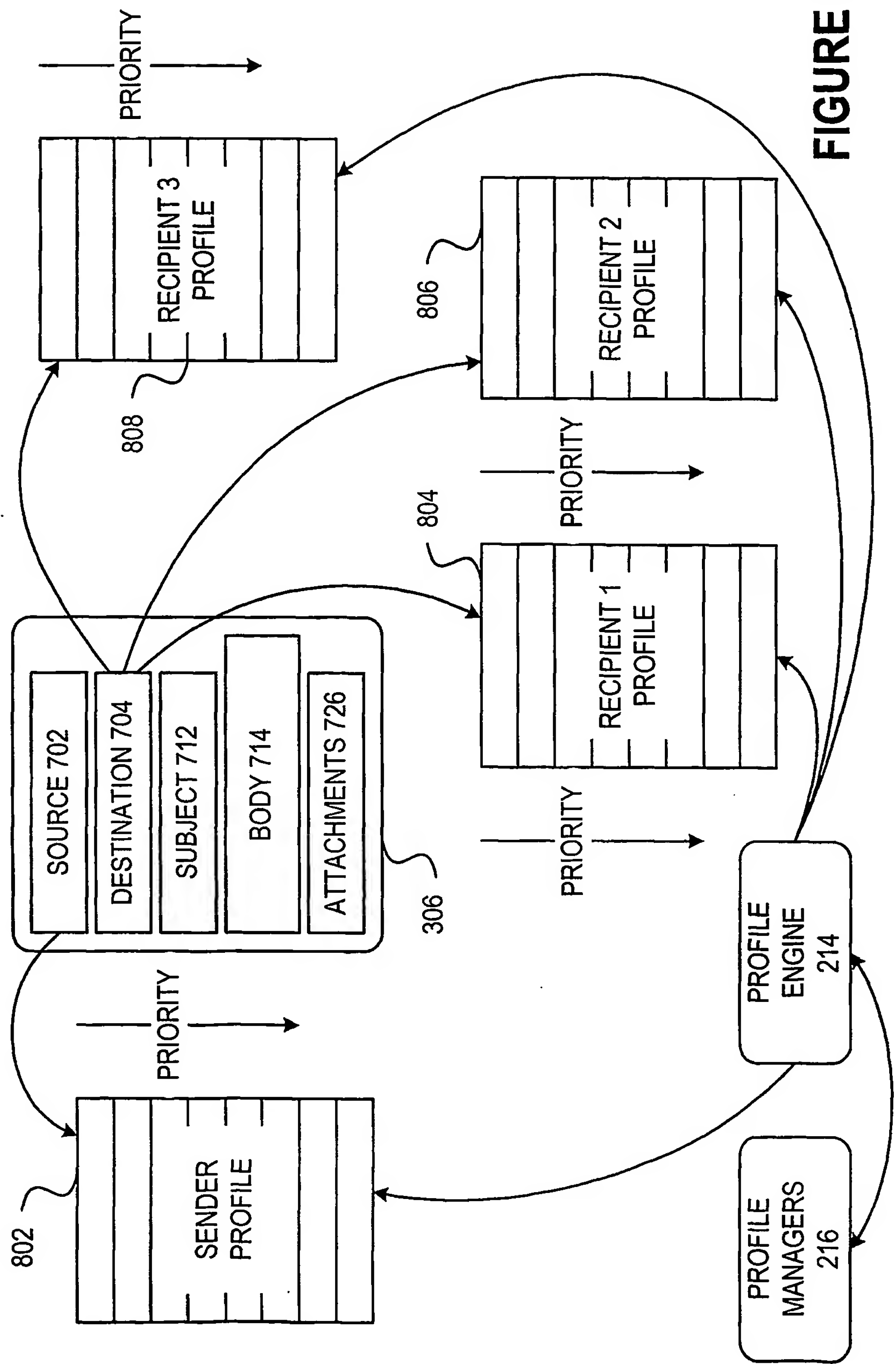


FIGURE 8

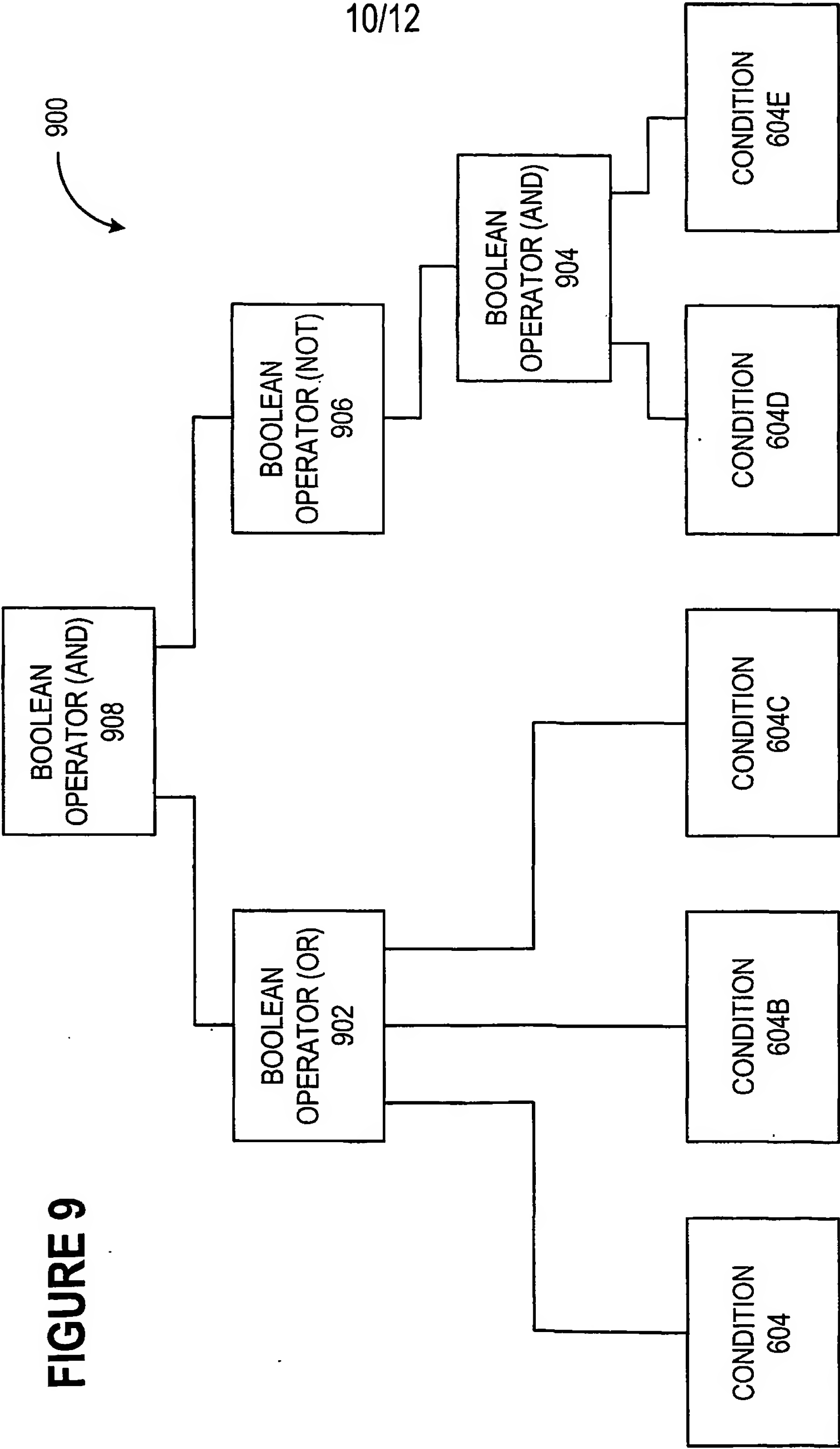


FIGURE 9

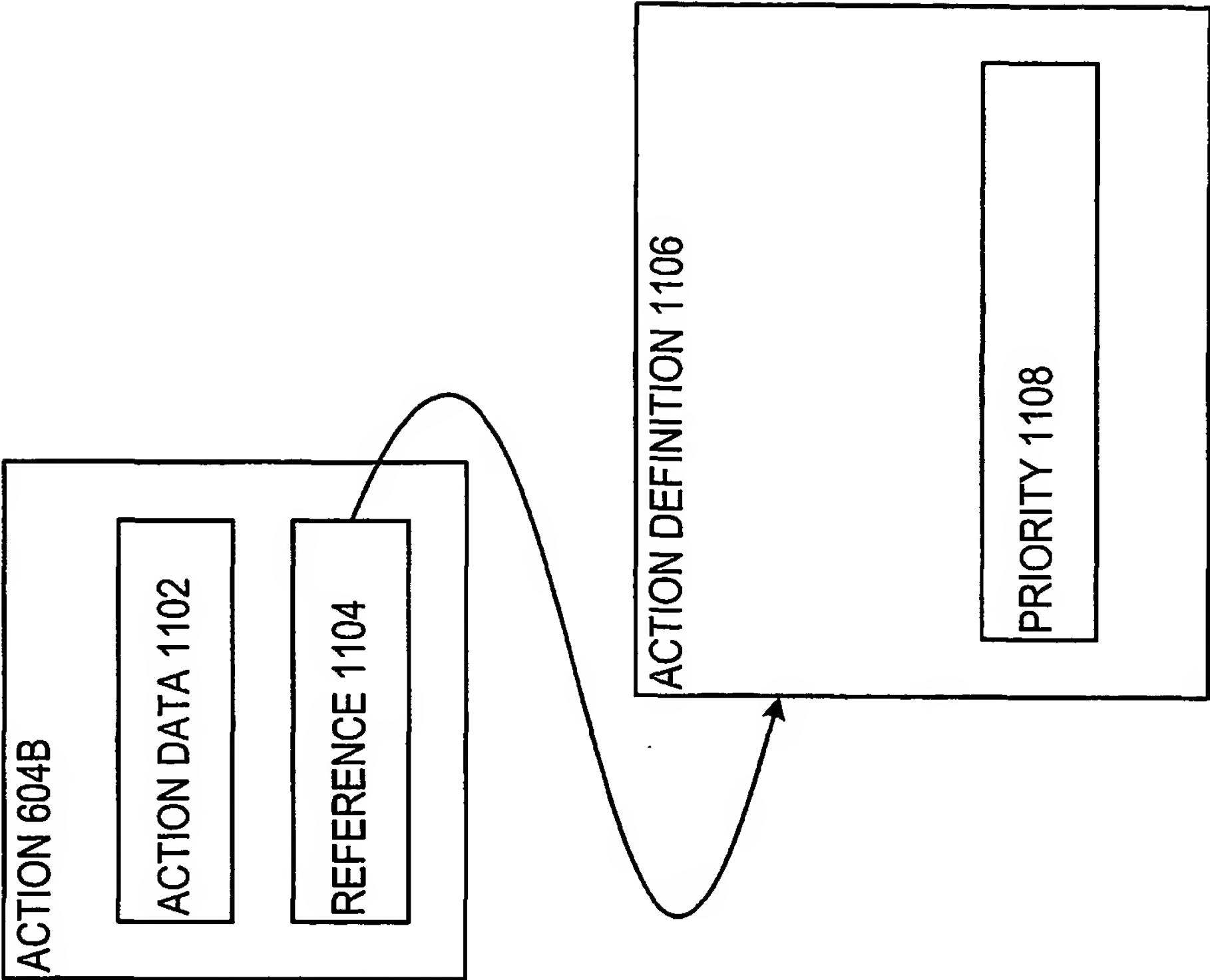


FIGURE 11

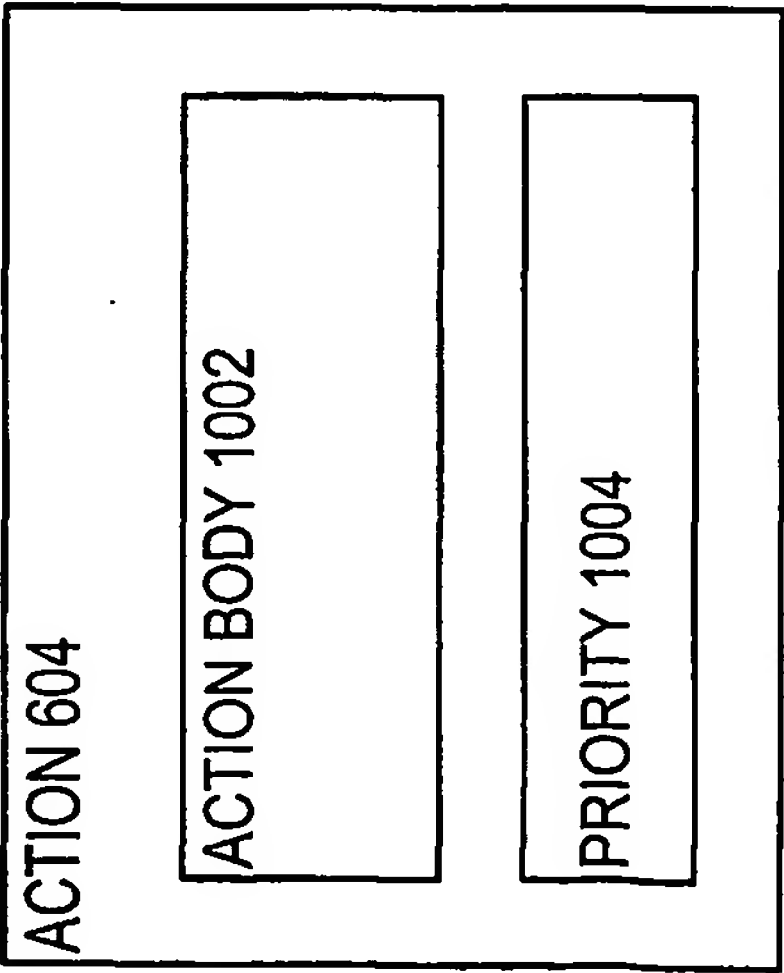
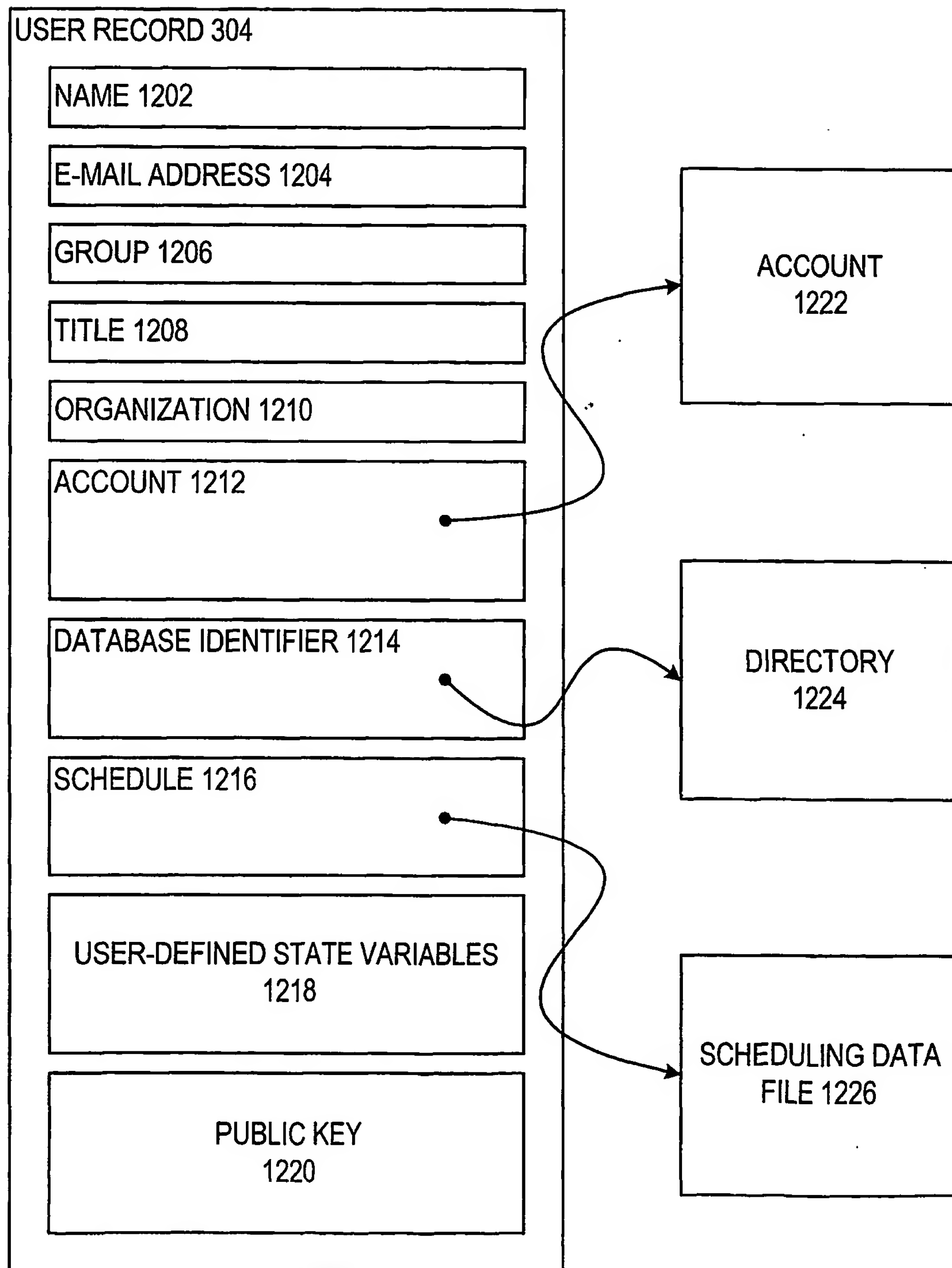


FIGURE 10

12/12

**FIGURE 12**